AD-A076 542 TRW DEFENSE AND SPACE SYSTEMS GROUP REDONDO BEACH CA F/G 5/1
AIRBORNE SYSTEMS SOFTWARE ACQUISITION ENGINEERING GUIDEBOOK FOR--ETC(U)
NOV 78 E JUCEVIC F33657-76-C-0677
UNCLASSIFIED TRW-30323-6010-TU-00 ASD-TR-79-5024 NL

1 OF 2
AD A 076542

LEVEL II 2

ASD-TR-79-5024

# Airborne Systems Software Acquisition Engineering Guidebook for CONFIGURATION MANAGEMENT

AD A076542

NOVEMBER 1978





PREPARED FOR
DEPUTY FOR ENGINEERING
AERONAUTICAL SYSTEMS DIVISION
WRIGHT-PATTERSON AFB, OH 45433



PREPARED BY
TRW DEFENSE AND SPACE SYSTEMS GROUP
ONE SPACE PARK
REDONDO BEACH, CA 90278

*NOTICE*

*When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.*

*This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.*

*This technical report has been reviewed and is approved for publication.*

JOHN M. HOEFERLIN, Project Engineer
Information Engineering Division

RICHARD J. SYLVESTER,
ASD Computer Resources Focal Point

*FOR THE COMMANDER*

GEORGE L. VON TERSCH, Colonel, USAF
Director of Avionics Engineering
Deputy for Engineering

*"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify ASD/ENAIA ,W-PAFB, OH 45433 to help us maintain a current mailing list".*

*Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.*

AIR FORCE/56780/1 October 1979 — 250

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| (19) REPORT DOCUMENTATION PAGE | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|
| 1. REPORT NUMBER (18) ASD-TR-79-5024 | 2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle) (6) Airborne Systems Software Acquisition Engineering Guidebook for Configuration Management | 5. TYPE OF REPORT & PERIOD COVERED (9) Final Reptis |
| | 6. PERFORMING ORG. REPORT NUMBER (14) TRW-30323-6010-TU-00 |
| 7. AUTHOR(s) (10) E. Jucevic (12) 138 | 8. CONTRACT OR GRANT NUMBER(s) (15) F33657-76-C-0677 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS TRW Defense and Space Systems Group One Space Park Redondo Beach, CA 90278 409 637 | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS PE64740F Project 2238 (16) |
| 11. CONTROLLING OFFICE NAME AND ADDRESS Hq ASD/ENAIA Wright-Patterson AFB OH 45433 (11) | 12. REPORT DATE Nov 78 / 13. NUMBER OF PAGES 112 |
| 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) | 15. SECURITY CLASS. (of this report) UNCLASSIFIED / 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release
Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Software Configuration Management
Software Acquisition Management
Configuration Identification, Configuration Control, Configuration Status Accounting

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report is one of a series of guidebooks which provide guidance in the acquisition management and engineering of Airborne Systems software procured under Air Force 800 series regulations. It provides specific guidance in the three basic CM areas: configuration identification, configuration control, and status accounting.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

409 637

## PREFACE

This guidebook is one of a series of guidebooks intended to assist the Air Firce Program Office and engineering personnel in software acquisition engineering for airborne systems. The contents of the guidebooks will be revised periodically to reflect changes in software acquisition policies and practices and feedback from users.

This guidebook has been prepared under the direction of the Aeronautical Systems Division (ASD), Deputy for Engineering (EN), in coordination with the Space and Missile Systems Organization (SAMSO), AirForce Systems Command (AFSC).

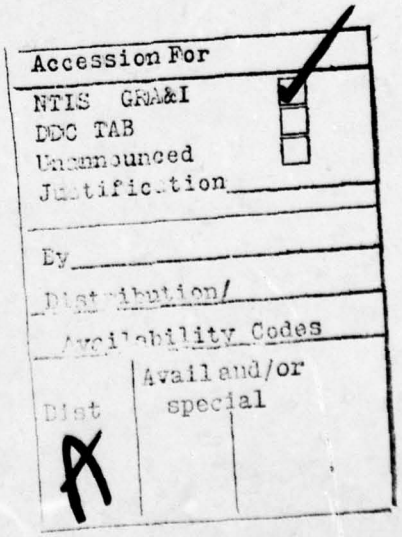
The series of Software Acquisition Engineering Guidebooks (Air-Borne Systems) is currently planned to cover the following topics:

Available Guidebooks

- Regulations, Specifications and Standards, ASD-TR-78-6; ADA058428
- Software Quality Assurance, ASD-TR-78-8; ADA059068
- Reviews and Audits, ASD-TR-78-7; ADA058429
- Statements of Work and Requists for Proposal, ASD-TR-79-5026
- Configuration Management, ASD-TR-79-5024
- Computer Program Documentation Requirements, ASD-TR-79-5025

Planned Guidebooks

- Verification, Validation and Certification ASD-TR-79-5028
- Requirements Analysis and Specification ASD-TR-79-5027
- Software Cost Analysis and Estimating
- Contracting for Software Acquisition
- SAE Guidebooks - Application and Use
- Computer Program Maintenance
- Software Development Planning and Control
- Software Testing and Evaluation
- Microprocessors and Firmware
- Software Development and Support Facilities



CONTENTS

CONTENTS (Continued)

CONTENTS (Concluded)

## ILLUSTRATIONS

## TABLES

## ABBREVIATIONS AND ACRONYMS

| | |
|---|---|
| ACI | Allocated Configuration Identification |
| AF | Air Force |
| AFLC | Air Force Logistics Command |
| AFR | Air Force Regulation |
| AFSC | Air Force Systems Command |
| ALC | Air Logistics Center |
| AMSDL | Acquisition Management Systems and Data Requirements List |
| ASD | Aeronautical Systems Division |
| ASPR | Armed Services Procurement Regulation |
| CCB | Configuration Control Board |
| CCBD | Configuration Control Board Directive |
| CDCN | Contract Document Change Notice |
| CDR | Critical Design Review |
| CDRL | Contract Data Requirements List |
| CI | Configuration Item |
| CII | Configuration Identification Index |
| CM | Configuration Management |
| CMO | Configuration Management Office |
| CMP | Configuration Management Plan |
| CO | Change Order |
| CPC | Computer Program Component |
| CPCI | Computer Program Configuration Item |
| CPCSB | Computer Program Configuration Sub-Board |
| CPDP | Computer Program Development Plan |
| CPIN | Computer Program Identification Number |
| CRISD | Computer Resources Integrated Support Data |
| CRISP | Computer Resources Integrated Support Plan |
| CRWG | Computer Resources Working Group |
| CSAR | Configuration Status Accounting Report |
| DID | Data Item Description |
| DMO | Data Manager Officer |
| DOD | Department of Defense |
| DODD | Department of Defense Directive |

| | |
|---|---|
| DODI | Department of Defense Instruction |
| DSARC | Defense Systems Acquisition Review Council |
| DUT | Document Update Transmittal |
| ECP | Engineering Change Proposal |
| FCA | Functional Configuration Audit |
| FCI | Functional Configuration Identification |
| FOT&E | Formal Operational Test and Evaluation |
| FQR | Formal Qualification Review |
| FQT | Formal Qualification Test |
| GFE | Government Furnished Equipment |
| HQ | Headquarters |
| ICD | Interface Control Drawing |
| ICN | Interface Change Notice or Installation Completion Notification |
| ICR | Interface Change Request |
| ICWG | Interface Control Working Group |
| IDS | Interface Design Specification |
| IFPP | Instructions for Preparation of Proposal |
| IOT&E | Initial Operational Test and Evaluation |
| MIL STD | Military Standard |
| NOR | Notice of Revision |
| O/S CMP | Operational/Support Configuration Management Procedures |
| OSD | Office of the Secretary of Defense |
| OT&E | Operational Test and Evaluation |
| PCA | Physical Configuration Audit |
| PCI | Product Configuration Identification |
| PCO | Procuring Contracting Officer |
| PDR | Preliminary Design Review |
| PM | Program Manager |
| PMD | Program Management Directive |
| PMP | Program Management Plan |
| PMRT | Program Management Responsibility Transfer |
| PO | Program Office |
| PQT | Preliminary Qualification Test |
| QA | Quality Assurance |
| RFP | Request for Proposal |

| | |
|---|---|
| RSS | Regulations, Specifications, and Standards |
| SAD | System Allocation Document |
| SAF | Secretary of the Air Force |
| SAMSO | Space and Missile Systems Organization |
| SCN | Specification Change Notice |
| SDR | System Design Review |
| SOW | Statement of Work |
| SRR | System Requirements Review |
| T&E | Test and Evaluation |
| TCP | Task Change Proposal |
| TCTO | Time Compliance Technical Order |
| TO | Technical Order |
| TRR | Test Readiness Review |
| USAF | United States Air Force |
| V&V | Verification and Validation |
| VDD | Version Description Document |
| WBS | Work Breakdown Structure |

# 1. INTRODUCTION

## 1.1 PURPOSE AND SCOPE

The purpose of this guidebook is to assist Air Force personnel in interpreting and applying the principles and requirements of Configuration Management (CM) to the acquisition of airborne system software (i.e., computer programs and related computer data).

General guidelines for software CM are presented, together with specific guidance in the three basic CM areas: configuration identification, configuration control, and configuration status accounting. Most attention is given to configuration control of the baseline items that are the primary concern of a procuring activity, but contractor internal configuration control of non-baselined items also is discussed because it is the level of control used for qualification testing (PQT and FQT) and because it is the nucleus of a contractor's baseline CM system.

AFSC's current CM policies and practices, as reflected by applicable Government regulations, specifications, and standards (RSS), provide the major directions for this guidebook. The most important of these RSS are summarized and are referenced frequently in the text. For CM planning, this guidebook relies on standard planning documents (PMP, CRISP, SOW, CDRL, CPDP, and CM Plan), and for contractor data items, it refers to standard Data Item Descriptions (DID's).

## 1.2 LIFE CYCLE RELATIONSHIPS

Software configuration management begins early in the software development life cycle and continues until the software is removed from the Government inventory. Procuring activity responsibility for software CM continues up to the time of Program Management Responsibility Transfer (PMRT), when the supporting command (normally AFLC) assumes it along with other program management responsibilities.

This guidebook concentrates on software CM requirements prior to PMRT, but covers the subsequent period briefly.

## 1.3 RELATIONSHIP TO OTHER GUIDEBOOKS

This guidebook relies on other volumes of this series for coverage of certain subjects related to CM:

a) Regulations, Specifications, and Standards Guidebook for methods of selecting and tailoring CM-related documents and referencing them in contracts.

b) Quality Assurance Guidebook for the quality assurance of CM programs.

c) Reviews and Audits Guidebook for planning and conduct of Functional Configuration Audits (FCA's), Physical Configuration Audits (PCA's), and Formal Qualification Reviews (FQR's).

d) Verification, Validation, and Certification Guidebook for verification and validation of CM-related activities and products.

e) Computer Program Documentation Requirements Guidebook for descriptions of configuration identification specifications and other CM-related documents and for guidance in the acquisition of CM-related documents.

## 1.4 CONTENTS OF THIS GUIDEBOOK

This guidebook has the following parts:

a. Section 1, Introduction. Describes the purpose and scope of the guidebook, states the general relationship of software CM to the system acquisition life cycle and to other volumes of this guidebook series, and outlines the contents.

b. Section 2, Relevant Documents. Surveys the Government regulations, specifications, and standards relevant to software CM on AFSC programs.

c. Section 3, General Guidelines for Software Configuration Management. Provides an overview of CM concepts and describes the variations that occur during a system life cycle and those required by program characteristics. Also outlines CM responsibilites on a program and offers some suggestions for CM planning and for monitoring contractor CM.

d. Section 4, Specific Guidance for Configuration Identification. Discusses system hierarchies, CPCI selection, configuration identification documents, specification trees, and item identifiers.

e. Section 5, Specific Guidance for Configuration Control. Defines items and periods of configuration control and discusses baseline configuration control, contractor internal configuration control, interface control, and control libraries.

f. Section 6, Specific Guidance for Configuration Status Accounting. Defines general responsibilities for configuration status accounting and aspects of both baseline and contractor status accounting. Also briefly discusses automated configuration status accounting.

g. Appendix A, Glossary. Defines basic software CM terms.

h. Appendix B, Outline for Contractor Software CM Plan. Presents outline for a contractor CM Plan suitable for AFSC software development procurements.

i. Appendix C, Bibliography of Government Documents. A list of Government regulations, specifications, and standards containing information pertinent to software CM.

j. Appendix D, Examples of Contractor Configuration Control Forms. Contains examples of five basic contractor configuration control forms: Design Problem Report (DPR), Documentation Update Transmittal (DUT), Software Problem Report (SPR), Software Modification Record (SMR), and Data Base Change Request (DBCR).

## 2. RELEVANT DOCUMENTS

Configuration management is widely discussed in DOD regulations, specifications, and standards (RSS). The most important of these RSS for ASD and SAMSO software acquisition programs are shown in the RSS tree in Figure 2-1. This tree begins at the top (box X) with a document that treats CM only as one of a number of disciplines involved in program management of an entire system acquisition. In the next box (Y), the documents are concerned with management of computer resources acquisition but still cover a multitude of subjects in addition to CM. Below that, Box A lists RSS that are devoted entirely to the subject of CM. The trunk then branches out into the three main functions of configuration management: configuration identification (B), configuration control (C), and configuration status accounting (D). A second level of branching shows three additional product development areas closely related to CM: configuration audits (E), transfer and turnover (F), and quality assurance for CM (G).

In each box of this RSS tree, the internal documents, which provide direction and guidance to Government participants, are separated from the military specifications and standards, which are instruments of contract compliance.

The purpose and scope of each document that appears in the RSS tree are described in Table 2-1. Comments concerning the specific mention of software CM in a document or the general applicability of a document to software CM also are included, as well as comments on the interrelationships of these RSS and on any pending RSS changes that are known or anticipated.

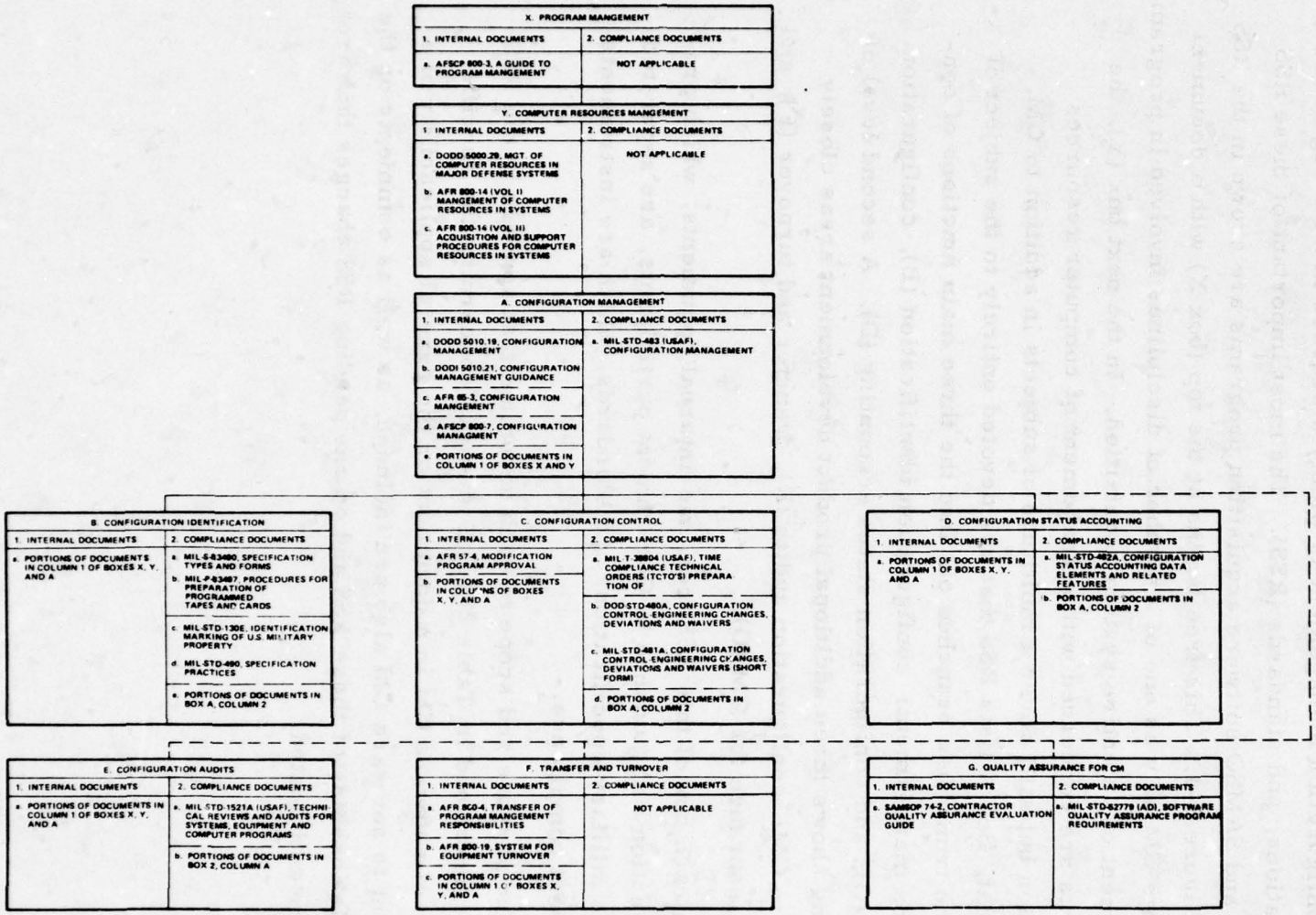



Figure 2-1. Configuration Management RSS Tree

## DOD DOCUMENTS

DOD Directive 5000.29, Management of Computer Resources in Major Defense Systems, April 26, 1976

a. Purpose. This directive establishes DOD policy for the management and control of computer resources during the development, acquisition, deployment, and support of major defense systems. Its principles also are intended to be applied to defense systems that do not fall in the "major acquisition" category.

b. Scope. Outlines DOD policy for requirements validation and risk analysis, configuration management of computer resources, computer resource life cycle planning, support software deliverables, milestone definition and attainment criteria, and software language standardization and control. Includes brief list of definitions of basic computer resource terms, the Charter of DOD Management Steering Committee for Embedded Computer Resources, and a memorandum on Technology Coordinating Papers (TCP's).

c. Comments. DOD computer resource CM policy is given in paragraph V.C as follows:

> Configuration Management of Computer Resources. Defense system computer resources, including both computer hardware and computer software, will be specified and treated as configuration items. Baseline implementation guidance for this action is contained in DOD Instruction 5010.21.

DOD Directive 5010.19, Configuration Management, July 17, 1968 (with Change 1, August 6, 1968; Change 2, April 7, 1970)

a. Purpose. This directive establishes DOD policies governing the configuration management of systems, equipments, and other designated materiel items.

b. Scope. Outlines DOD policy for configuration identification, configuration control, configuration status accounting, and overall CM activities. Includes list of CM definitions. References include DOD Instruction 5010.21.

c. Comments. This directive makes no specific mention of computer resources or software, but is totally applicable to software acquisition programs.

## USAF DOCUMENTS

AF Regulation 57-4, Modification Program Approval, 15 December 1977 (with Interim Message Change 78-1, 9 February 1978).

a. Purpose. This regulation defines Air Force policies and procedures for planning, documenting, and obtaining approval of modifications to correct deficiencies in or improve the capabilities of existing Air Force equipment and nonnuclear munitions for which the Air Force has logistic support responsibility. It implements the configuration control portions of AFR 65-3 that pertain to modifications.

b. Scope. This regulation defines policies and responsibilities for the Air Force modification program and describes procedures for justifying, submitting, and approving modifications. It also describes the documents involved in the modification program.

c. Comments. This regulation supersedes AF Regulation 57-4 dated 27 January 1972 and titled "Retrofit Configuration Changes."

AF Regulation 65-3, Configuration Management, 1 July 1974 (with Change 1, 1 September 1974; AFSC Supplement 1, 25 July 1975)

a. Purpose. This document is a Joint DOD Services/Agency Regulation that prescribes uniform policies and guidance for DOD components responsible for implementation of configuration management within DOD. The document carries a different identifier for each service. AFR 65-3 is the Air Force identifier. A series of appendices provide implementing instructions for the various services and agencies. The Air Force appendix is F (added via Change 1).

b. Scope. Prescribes policies and guidance for configuration identification, configuration control, configuration status accounting, and configuration audits. Includes list of CM definitions.

c. Comments. Is generally applicable to CM of software, but includes only a few specific references to "computer programs." Chapter 6 of AFR 800-14, Volume II, provides guidance in applying the CM practices and procedures of AFR 65-3 to computer resources.

AF Regulation 800-14, Volume I, Management of Computer Resources in Systems, 12 September 1975 (with AFSC Supplement 1, 8 August 1977)

a. Purpose. This volume of AFR 800-14 establishes Air Force policy for the acquisition and support of computer equipment and computer programs employed as dedicated elements, subsystems, or components of systems developed or acquired under the management concept established in AFR 800-2.

b. Scope. Lists a dozen short policy statements, including one on configuration management (item A.3.j) and one on the requirements of Program Management Directives (PMDs) and the provisions of Program Management Plans (PMPs). Also defines responsibilities of HQ USAF, AFSC, AFLC, ATC, the program manager, using activities, and the Air University. Includes a glossary of glossary of terms. Included in AFSC Supplement 1 are policy statements on microprocessors, microcomputers, and firmware and definitions of these three items and of verification and validation.

c. Comments. Is very explicit on policy for computer program acquisition and support.

AF Regulation 800-14, Volume II, Acquisition and Support Procedures for Computer Resources in Systems, 26 September 1975 (with AFLC Supplement 1, 18 October 1976)

a. Purpose. This volume of AFR 800-14 describes procedures for implementing the policies of Volume I and other related publications as these policies pertain to the acquisition and support of computer resources.

b. Scope. This volume consolidates and explains the applicability of a large group of regulations, specifications, and manuals to the acquisition and support of computer resources. It restates portions of these other documents and amplifies the policies contained there. It includes chapters on computer resources in the system acquisition life cycle, planning, engineering management, testing of computer programs, configuration management, documentation, identifying contractual requirements, turnover and transfer, and support during deployment. The Configuration Management chapter (6) provides guidance in applying the CM practices and procedures of AFR 65-3 to computer resources throughout the system acquisition life cycle. AFLC Supplement 1 to Volume II includes a chapter (10,3) on the USAF Computer Program Identification Numbering (CPIN) System.

c. Comments. Has many specific references to software.

## Table 2-1. Description of CM-Related Regulations, Specifications, and Standards (Sheet 1 of 2)

| ERNMENT INTERNAL DOCUMENTS (NON-CONTRACTUAL) | |
|---|---|
| | AFSC DOCUMENTS |
| ly 17, 1968<br>70)<br>governing the<br>and other<br><br>ification,<br>ag, and overall<br>ferences<br><br>ion of computer<br>software | DOD Instruction 5010.21, Configuration Management Implementation Guide, August 6, 1968 (with Change 1, January 29, 1969)<br><br>a. Purpose. This instruction provides guidance for the implementation of the CM policies established by DOD Directive 5010.19, "Configuration Management," July 17, 1968.<br><br>b. Scope. Provides guidance in configuration identification, configuration control, configuration status accounting, configuration audits, governing documentation, contract provisions, and logistic support aspects of CM. Enclosure 1 lists the same definitions listed in DOD Directive 5010.19. Also has enclosures on "Item Numbering" and "Configuration Status Accounting Data Content."<br><br>c. Comments. This instruction specifies its applicability to "operational computer programs." | AFSC Pamphlet 800-3, A Guide for Program Management, 9 April 1976<br><br>a. Purpose. This pamphlet describes the general subjects involved in managing the acquisition of Air Force systems and associated elements. It is intended to assist program managers, program office personnel, and others involved in the acquisition process to understand the process better and to help them in planning and accomplishing their assigned functions and responsibilites.<br><br>b. Scope. In its first five chapters, this large pamphlet (more than 200 pages) describes the acquisition process from conception through deployment, identifying key events and activities normally occurring during each phase. Chapters 6 through 21 discuss major subjects involved in managing acquisition programs, including configuration management, interface management, engineering management, data management, test and evaluation, deployment management, and program office organization. The pamphlet also contains two attachments on the preparation of Program Management Plans (PMPs).<br><br>c. Comments. The present version has very few specific references to computer resources or software but is generally applicable to software development programs. AFSC/XRF is preparing a new chapter that will specifically discuss computer resource management in a program office. |
| | | AFSC Pamphlet 800-7, Configuration Management, 1 December 1977<br><br>a. Purpose. This pamphlet describes CM concepts, techniques, and procedures for the guidance of AFSC personnel who are responsible for applying CM to systems, system modifications, system segments, equipments, computer programs, and other designated material items referred to as configuration items (CIs).<br><br>b. Scope. Describes the details of configuration identification, configuration control, and configuration status accounting and includes chapters on reviews and audits and computer program CM. Contains new material as well as considerable material from AFSCM/AFLCM 375-7, which it supersedes. Retained from 375-7 (now in subsection 6-6) is an explanation of how MIL-STD-483 appendices apply to the acquisition of computer program configuration items.<br><br>c. Comments. The amount of specific detail on software CM, combined with a lot of specific information on CM functions, makes this the most useful single document currently available for AFSC software CM planners and managers. |
| 1974 (with<br>ly 1975)<br>gency Regu-<br>for DOD<br>guration man-<br>rent identifier<br>ifier. A series<br>the various<br>F (added via<br><br>ration identi-<br>accounting,<br>ions.<br><br>re, but<br>e programs."<br>ance in apply-<br>computer | AF Regulation 800-4, Transfer of Program Management Responsibility 10 March 1975 (with AFSC/AFLC Supplement 1, 14 August 1975; SAMSO Supplement 1, 6 April 1976).<br><br>a. Purpose. This regulation states USAF policy and assigns responsibility for Program Management Responsibility Transfer (PMRT), which is the transfer of program management responsibility from the implementing command (AFSC) to a supporting command (normally AFLC).<br><br>b. Scope. This two-page regulation briefly explains terms and defines policy and responsibilities associated with transfer of program management. A Transfer Working Group (TWG) is required. The five-page AFSC/AFLC Supplement 1 adds a number of new terms, expands the policy and responsibility sections, and outlines a PMRT plan. | |
| | | SAMSO DOCUMENT |
| rt Procedures<br>(with AFLC<br><br>ocedures for<br>ated publica-<br>support of<br><br>pplicability of<br>mals to the<br>restates por-<br>icies contained<br>in the system<br>ment, testing<br>documentation,<br>ransfer, and<br>agement<br>ctices and<br>ughout the<br>Volume II<br>gram Identi-<br><br>re. | AF Regulation 800-19, System or Equipment Turnover, 27 May 1975.<br><br>a. Purpose. This regulation establishes policy and principles for the efficient turnover to an operating command of systems or equipments developed under the program management concept established in AFR 800-2.<br><br>b. Scope. Explains terms and defines policy, responsibilities, and turnover documentation. Includes a list of turnover principles based on past problems, including some related specifically to computer programs. | SAMSO Pamphlet 74-2, Contractor Quality Assurance Evaluation Guide, 1 September 1976.<br><br>a. Purpose. This pamphlet provides guidance to personnel responsible for the evaluation of a contractor's software quality program when MIL-S-52779 (AD) is invoked in the contract.<br><br>b. Scope. Provides a complete paragraph-by-paragraph analysis of MIL-S-52779, first repeating the paragraph verbatim, then giving descriptions and examples of practices applied by contractors in the past, and finally listing evaluation criteria. In the CM area, the emphasis is on having an appropriate CM plan and performing and documenting periodic audits of the CM function. The library controls portion emphasizes establishment of a computer program library and controlling the source and object program materials. For reviews and audits, the main concerns are establishment of a review and audit schedule and compliance of procedures with MIL-STD-1521. |

## SPECIFICATIONS

MIL-P-83497 (USAF), Procedures for Preparation of Programmed Tapes and Cards, 16 August 1976.

a. Purpose. This specification covers the general requirements for preparing, handling, and storing magnetic tapes, punched tapes, and punched cards for *organizational, intermediate, or depot level* use in conjunction with embedded computer systems in operational flight programs, test equipment, simulators, command and control systems, and communications-electronic-meteorological *equipment.*

b. Scope. This specification defines the size, composition, and other characteristics of tapes and cards; methods of marking identification on these media; quality assurance provisions; and preservation, packaging, and packing. It references three ANSI tape specifications and several Federal and Military specifications concerning packaging and marking.

c. Comments. This specification is cited in AFSCP 800-7 (paragraph 6.4.h(2)) and in MIL-STD-130E as the contractually applicable authority for identifying and marking or labeling tapes and cards. *Note, however, that MIL-P-83497* goes beyond marking, packaging, etc., and actually defines the physical characteristics of tapes and cards.

MIL-S-52779 (AD), Software Quality Assurance Program Requirements, 5 April 1974.

a. Purpose. This specification requires establishment and implementation of a Software Quality Assurance (QA) Program by the contractor to assure that software delivered complies with the requirements of the contract.

b. Scope. Briefly defines QA requirements for software configuration management, library controls, reviews and audits, and other software development areas.

c. Comments. Interpretation of MIL-S-52779 requirements is provided by SAMSO Pamphlet 74-2.

MIL-S-83490, Specificat

a. Purpose. This spec
partition of program
agencies.

b. Scope. Briefly stat
types of specificatio
and identifies fifteen
requirements and in

(1) Form 1, Specif

o Form 1a, wi

o Form 1b, wi

(2) Form 2, Specif
Military Requir

(3) Form 3, Specif

In addition, MIL-S-
cable to specificatio
ment documents.

c. Comments.

(1) MIL-STD-490 d
types listed in

(2) MIL-STD-483,
specifications f
operational inve

## STANDARDS

MIL-STD-130E, Identification Marking of U.S. Military Property, 5 August 1977

a. *Purpose.* This standard establishes the item marking requirements and methods for identification of items of DOD military property.

b. Scope. States both general and detail marking requirements.

c. Comments. Paragraphs 5.2.1.2 (Programmed Tapes and Cards) and 5.5 (Exclusions) refer to MIL-P-83497 as the contractually applicable authority for identifying and marking or labelling tapes and cards.

DOD-STD-480A, Configuration Control - Engineering Changes, Deviations and Waivers, 12 April 1978.

a. Purpose. This standard provides requirements for maintaining configuration control of configuration items. It is intended to be imposed on contractors who are able to provide the Government with most of the information needed to properly evaluate *the merits of complex engineering changes* that may involve changes to related configuration items.

b. Scope. *This standard provides* detailed requirements for (1) preparing and submitting engineering change proposals (ECPs), deviations, waivers, and notices of revision (NORs), (2) submitting the technical, fiscal, and logistic supporting information necessary to define the impact of a proposed engineering change, and (3) submitting the information necessary to maintain the current status of configuration identification. DOD-STD-480A covers a broader area than its companion standard, MIL-STD-481, and requires a more complete analysis of the impact of engineering changes. It contains a list of definitions based on DODD 5010.19 and other sources.

c. Comments.

(1) DOD-STD-480A supersedes MIL-STD-480. Most changes in the new standard are based on the *software configuration control requirements of* MIL-STD-483, Appendix XIV. The major changes are: (a) the definition of Class I changes in paragraph 4.2.1 incorporates the MIL-STD-483 requirements and *adds some details;* (b) the Appendix A instructions for preparing ECP forms incorporates some MIL-STD-483 requirements (Blocks 5a, 14, 15, 29, and 31 and page 4), changes one (page 6), and *omits some* (Blocks 5e, 8, 18, and 21 and page 5); (c) a figure showing life applications of ECP form pages is renumbered Figure 1 and now includes conceptual and validation phases instead of a contract definition phase; and (d) the processing target for routine ECP's is changed from 45 days to whatever the application requires. *DOD-STD-480A does not* include the MIL-STD-483 stipulation that Notices of Revision (NOR's) are to be used for CPCIs only when the CPCIs are off-the-shelf items.

(2) In cases of conflict between DOD-STD-480A and MIL-STD-483, this guidebook assumes that MIL-STD-483 takes precedence for AFSC software applications. When these standards are used as contract compliance documents, however, they should be tailored to meet a program's specific requirements.

(3) AFSC Supplement 1 (paragraph 3-5a) to AFR 65-3 says that MIL-STD-480 should be used in preference to MIL-STD-481 unless MIL-STD-480 would *impose an undue burden on the* contractor. It is assumed that this requirement now applies to DOD-STD-480A instead of MIL-STD-480. For other RSS references to MIL-STD-480, the same substitution probably is *valid but each case* should be examined separately when contract compliance is involved.

*MIL-STD-481A, Config*
Waivers (Short Form),

a. *Purpose.* This sta
mittal of abbreviate
waivers. It is inte
not peculiar to spec
expected to know all
standard, the procu

b. Scope. MIL-STD-4
DOD-STD-480A. It
described in DOD-S
simplified.

c. Comments.

(1) MIL-STD-481A
applied to softw

(2) AFSC Supplem
should be used
an undue burde
now applies to

MIL-STD-483 (USAF), Configuration Management Practices for Systems, Equipment, Munitions, and Computer Programs, 31 December 1970 (with Notice 1, *1 June 1971).*

a. Purpose. This standard establishes uniform configuration management practices that can be tailored to all USAF systems and configuration items, including computer programs.

b. Scope. This standard is a loosely structural collection of material on a wide variety of CM subjects. It does not cover all major topics of CM because it is intended to be used with a series of DOD standards on CM-related subjects: MIL-STD-480, 481, 482, and 490. Part 1 of this standard states general requirements for CM and explains the application of the fifteen appendices in Part 2. These appendices include important computer program topics, topics with general applicability to both software and hardware, and topics not applicable at all to computer programs. *Especially useful for software* acquisition are Appendices II (Interface Control), VI (Computer Program Configuration Item Specification; i.e., Part I and Part II specifications), VIII (Specification *and Support Documentation Maintenance,* Computer Program; discusses SCNs, VDDs, configuration indexes, and change status reports), IX (Document and Item Identification Numbering and Marking), and XIV (Engineering Changes, *Computer Programs;* discusses Class I and II changes, ECPs, and ACSNs).

c. Comments. MIL-STD-483 has more than its share of opaque text. Nevertheless, it is the most important document available for contract application of software CM requirements and, in addition, provides useful insights into CM practices. MIL-STD-483 may be replaced by a DOD-level standard whose draft version is identified as MIL-STD-XXX, and some of its appendices on specifications may be incorporated into a revision of *MIL-STD-490.*

MIL-STD-490, Specific
1 February 1969)

a. Purpose. This st
program-peculiar
It complies with th
Directive 5010.19

b. Scope. The body
general and for ea
devoted to the deta
including the follo

a. Type A, Syste

b. Type B5, Con

c. Type C4, Inve

d. Type C5, Con

c. Comments.

(1) For the four
supplemented
requirements
MIL-STD-49
included in M
crepancies b

(2) Notice 2 appl

# Table 2-1. Description of CM-Related Regulations, Specifications, and Standards (Sheet 2 of 2)

E DOCUMENTS FOR CONTRACTUAL APPLICATIONS

SPECIFICATIONS

ments,

mentation of a
assure that
ract.

ration manage-
e development

rovided by

MIL-S-83490, Specifications, Types and Forms, 30 October 1968

a. Purpose. This specification prescribes general requirements for the preparation of program-peculiar specifications required by DOD departments or agencies.

b. Scope. Briefly states the required contents and intended uses of five general types of specifications (system, development, product, process, and material) and identifies fifteen specific types of specifications. It also states the requirements and intended uses of the following "forms" of specifications:

(1) Form 1, Specifications to Military Standards

- o Form 1a, with maximum formal control
- o Form 1b, with limited formal control

(2) Form 2, Specifications to Commercial Practices, with Supplemental Military Requirements

(3) Form 3, Specifications to Commercial Practices

In addition, MIL-S-83490 briefly states quality assurance provisions applicable to specifications and lists information items required in data procurement documents.

c. Comments.

(1) MIL-STD-490 describes the detailed format for the fifteen specification types listed in MIL-S-83490.

(2) MIL-STD-483, paragraph 3.4.8, requires form 1a or 1b to be used for specifications for new design configuration items to be produced for the operational inventory, unless the procuring activity specifies otherwise.

MIL-T-38804 (USAF), Time Compliance Technical Orders (TCTOs), Preparation of, 31 July 1972 (with Amendment 2, 1 November 1977).

a. Purpose. This specification identifies the requirements for preparing Time Compliance Technical Orders (TCTOs) for use in, among other things, announcing retrofit computer program changes affecting weapon systems, automatic test equipment, simulators, and on-board command and control systems. TCTOs also are used to impose or direct usage restrictions and to order retesting, special one-time inspection, or replacement.

b. Scope. MIL-T-38804 defines in detail the form and content of immediate action, urgent action, and routine actions TCTOs and their supplements. Amendment 2 adds specific requirements for TCTOs that involve computer program changes.

c. Comments. According to AFSCP 800-7, paragraph 6-6.0, TCTOs are used to issue all retrofit engineering changes for both CPCIs and CIs and are released, installed, and reported according to AFR 57-4. However, TO 00-5-1, "AF Technical Order System," dated 15 August 1977, says that the AF TO system does not apply to "computer programs and support documentation managed in accordance with AF Regulation 800-14." It is not clear whether the latter statement really excludes the use of TCTOs in AFR 800-14 programs. In any case, AFSCP 800-7 appears to have precedence on this matter. For new procurements, additional guidance on the use of TCTOs should be obtained from appropriate AFSC sources.

STANDARDS

viations and

g configuration
contractors
rmation needed to
that may involve

preparing and
waivers, and
l, and logistic
oposed engineer-
aintain the cur-
vers a broader
es a more com-
ins a list of

e in the new
requirements of
a) the definition
IL-STD-483
instructions for
equirements
(page 6), and
figure showing
re 1 and now
ntract definition
changed from
-480A does not
ation (NOR's) are
lf items.

D-483, this guide-
FSC software
compliance
rogram's specific

hat MIL-STD-480
L-STD-480 would
d that this
IL-STD-480. For
tion probably is
contract com-

MIL-STD-481A, Configuration Control - Engineering Changes, Deviations and Waivers (Short Form), 18 October 1972.

a. Purpose. This standard provides requirements for the preparation and submittal of abbreviated engineering change proposals (ECPs) and deviations and waivers. It is intended for use in contracts involving multi-application items not peculiar to specific systems or with contractors who cannot reasonably be expected to know all of the consequences of engineering changes. With this standard, the procuring activity performs most of the impact analysis.

b. Scope. MIL-STD-481A is a smaller and simpler document than DOD-STD-480A. It calls for a one-page ECP instead of the six-page ECP described in DOD-STD-480A. Deviation and waiver requirements also are simplified.

c. Comments.

(1) MIL-STD-481A does not address itself specifically to software but can be applied to software.

(2) AFSC Supplement 1 (paragraph 3-5a) to AFR 65-3 says that MIL-STD-480 should be used instead of MIL-STD-481 unless MIL-STD-480 would impose an undue burden on the contractor. It is assumed that this requirement now applies to DOD-STD-480A instead of MIL-STD-480.

MIL-STD-482A, Configuration Status Accounting Data Elements and Related Features, 1 April 1974.

a. Purpose. This standard provides a comprehensive listing of standard data elements to be used in tailoring the content of configuration status accounting records and reports, in accordance with DODD 5010.19 and DODI 5010.21.

b. Scope. This standard prescribes status accounting standard data elements and interim (nonstandard) data elements and their related data items, codes, use identifiers, data chains, and field lengths. It does not prescribe data elements to be used nor does it specify the format or frequency of reports. Such requirements are to be specified by the managing organization.

c. Comments. This standard is applicable to either automated or manual status accounting systems.

MIL-STD-490, Specification Practices, 30 October 1968 (with Notice 1, 1 February 1969)

a. Purpose. This standard establishes uniform practices for the preparation of program-peculiar specifications required by DOD departments or agencies. It complies with the configuration identification concepts established by DOD Directive 5010.19 and DOD Instruction 5010.21.

b. Scope. The body of MIL-STD-490 states requirements for specifications in general and for each of the six sections of a specification. Appendices are devoted to the detailed requirements of 15 different types of specifications, including the following 4 types applicable to computer programs:

a. Type A, System Specification

b. Type B5, Computer Program Development Specification

c. Type C4, Inventory Item Specification

d. Type C5, Computer Program Product Specification

c. Comments.

(1) For the four types of specifications listed above, this standard should be supplemented by MIL-STD-483 (USAF) which states the specification requirements in greater detail. A proposed revision, identified as MIL-STD-490A, may contain some of the specification requirements now included in MIL-STD-483 (USAF). It also is supposed to resolve discrepancies between MIL-STD-483 and MIL-STD-490.

(2) Notice 2 applies only to the Army.

MIL-STD-1521A (USAF), Technical Reviews and Audits for Systems, Equipments, and Computer Programs, 1 June 1976.

a. Purpose. This standard prescribes requirements for the conduct of technical reviews and audits on systems, equipments, and computer programs.

b. Scope. This standard identifies contractor and Government responsibilities in the conduct of reviews and audits and outlines the minimum items of information to be presented during the following seven technical reviews and audits:

System Requirements Review (SRR)

System Design Review (SDR)

Preliminary Design Review (PDR)

Critical Design Review (CDR)

Functional Configuration Audit (FCA)

Physical Configuration Audit (PCA)

Formal Qualification Review (FQR)

The FCA and PCA are CM-oriented activities in that they verify product conformance to specifications and other contract requirements. The FQR also is considered a configuration audit if it is included in the Configuration Item Development Record of the CPCI Configuration Index. The FQR is primarily a system engineering activity focused on the product itself and the other four reviews are entirely so.

c. Comments. Additional information on all seven reviews and audits is provided in AFSCP 800-7, and the FCA, PCA, and FQR are discussed in Appendix XII of MIL-STD-483 (USAF).

# 3. GENERAL GUIDELINES FOR SOFTWARE CONFIGURATION MANAGEMENT

This section presents general information on the planning and implementation of configuration management. It outlines the basic concepts of software CM and the major variations that occur during development. It also discusses software CM responsibilities of Government agencies and contractors and the planning and monitoring of software CM.

## 3.1 BASIC CONCEPTS OF SOFTWARE CM

A major management problem in software product development is the need to control changes to the evolving product. These changes must be controlled and documented in such a way that the documents reflecting different points of view of the product remain compatible with each other and with the product itself. The design specification must comply with the requirements specification, and the evolving product must comply with both of these. Furthermore, the test procedures must be based on the product versions to be tested, not on earlier ones. Compatibility in the reverse direction also must be maintained when requirements, design, or test documents are affected by problems found in the actual product.

Configuration management provides an effective means to achieve these goals. In the words of AFR 65-3, configuration management is "a discipline applying technical and administrative direction and surveillance to:

(1) identify and document the functional and physical characteristics of a configuration item,

(2) control changes to those characteristics, and

(3) record and report change processing and implementation status."

The three CM areas that produce these results are configuration identification, configuration control, and configuration status accounting. A fourth CM area, configuration audits, verifies that a completed product and its documents meet contractual requirements.

### 3.1.1 Basic Concepts of Configuration Identification and Baselines

Configuration identification is established in the form of technical documentation that becomes more detailed as development proceeds. Three stages of configuration identification generally are employed during the development of a system:

a) Functional Configuration Identification (FCI). First, the functional configuration identification for the system is defined in the System Specification or System Segment Specification, which are performance-oriented requirements documents.

b) Allocated Configuration Identification (ACI). A little later in the development process, the requirements of the functional configuration identification are allocated to the CPCIs (computer program configuration items) and hardware CIs of the system in a series of Development Specifications. (For a CPCI or hardware CI that is not being developed as part of a larger item, the Development Specification becomes the functional configuration identification and an allocated configuration identification is not necessary.)

c) Product Configuration Identification (PCI). Finally, the product configuration identifications of the system CPCIs and hardware CIs are defined in a series of Product Specifications that describe the as-built configurations.

When one of these three configuration identifications is "baselined," it becomes the basis for both configuration control and status accounting for that configuration. Baselining is placing a configuration identification under the procuring activity's contractual configuration control system. Baselines are established at strategic points in a system acquisition life cycle to ensure that completed development work is adequately controlled during subsequent phases.

The three system development baselines defined in Government CM documents are:

a) Functional Baseline. Established by procuring activity program office authentication of the functional configuration identification (i.e., System or System Segment Specification), normally after the first, or conceptual, phase of the system acquisition life cycle is completed and the second, or validation, phase is in progress.

b) Allocated Baseline. Established by program office authentication of the allocated configuration identification (i.e., Development Specifications) at the end of the second, or validation, phase or early in the third, or full-scale engineering development, phase.

c) Product Baseline. Established by program office authentication of the product configuration identification (i.e., Product Specifications) near the end of the third, or full-scale engineering development, phase.

At any point in the system life cycle, a controlled item's current configuration is defined by its baselined identification documents plus all approved changes to those documents.

There is a definite distinction between configuration identification and baselines. As stated in AFSCP 800-7, "Identification is used for visibility and baselines are used for control. Identification becomes a baseline only when designated and fixed at a specific time as a reference point for change control." (AFSCP 800-7, paragraph 3.a(2); similar wording appears in paragraph 1-5g of AFSC Supplement 1 to AFR 65-3.)

As defined in Government CM documents and as discussed above, the word "baseline" means a controlled configuration identification document or set of such documents, but it is also commonly used to refer to the points in the life cycle where the different stages of baseline configuration control begin.

The term "configuration identification" also has a secondary meaning that is logical and useful. By DOD definition, "configuration identification" is a document or set of documents that defines the configuration of an item. In this sense, it represents one or more material things (documents). As a part of configuration management, however, it is grouped with two things that are not material things but are operating processes: configuration control and status accounting. It is natural,

therefore, when discussing the functions of CM to expand the scope of the term "configuration identification" to that of a third process that performs all tasks associated with identification of an item's configuration, including identification of the CPCIs in a system and assignment of unique item identifiers to software and documents.

### 3.1.2 Basic Concepts of Configuration Control

Configuration control is the major process of CM. It is the process by which change decisions are made (by the Configuration Control Board, or CCB), administered (by the Configuration Management Office, or CMO), and implemented (by software assembly and maintenance personnel).

Sometimes configuration control is called "change control," which really describes the process more accurately. In Government CM documents, "configuration control" is the term used in definitions and descriptions of the control process while "change control" appears elsewhere occasionally.

The decision-making part of configuration control determines whether proposed changes to a controlled document or software item will be beneficial to the Government in terms of operational effectiveness, support needs, cost, or schedule. The change administration and implementation parts ensure that all approved changes to a configuration are properly incorporated in the affected documents and software code and that no other changes find their way in.

The major steps in the configuration control process are as follows:

a) The need for a change to a baselined document or software item arises because of a change in requirements or because of a problem.

b) The need for the change is documented in a change proposal.

c) The CMO logs and numbers the change proposal.

d) The CCB reviews the change proposal and decides whether to approve the change, disapprove it, or obtain additional information. If the CCB lacks the authority to approve certain change proposals, it submits them to a higher authority.

e) The CCB assigns an approved change proposal to the appropriate organization for implementation.

f) The proposed change is implemented in the baselined item.

g) The CMO closes the change cycle for this change.

At the end of each such change cycle, the related documents and software files should be compatible with each other; that is, they should reflect exactly the same software configuration.

During the early stages of configuration control (see Figure 3-1), compatibility is not a problem, because only the functional configuration identification (i.e., System Specification) is controlled. After the allocated configuration identification documents (Development Specifications) are baselined, compatibility between the two levels of baselined documents is required. When the product baseline is achieved, the Product Specifications and the CPCIs themselves also become subject to configuration control and its compatibility requirements. All three baselines are controlled throughout the lifetime of the system. The configuration control of these baselines is referred to in this guidebook as "baseline configuration control."

Software items and documents that are not baselined are not subject to baseline configuration control, but may be placed under contractor internal configuration control during certain periods of development. (See Figure 3-1.) This applies both to items that will be baselined later in the development process, such as CPCIs and Product Specifications, and to items that will never be baselined, such as User Manuals, test documents, and non-deliverable software items. Contractor internal configuration control is almost entirely a contractor activity whereas baseline configuration control, because of its direct contractual effects, heavily involves both contractor and procuring activity. Government documents on CM occasionally allude to contractor internal configuration control but do not offer much help in understanding its needs and practices or guidance in evaluating its plans or performance. In the Government documents, "configuration control" generally means baseline configuration control.

Government CM documents advise procuring activities to allow contractors to use their existing CM systems and caution them about imposing unnecessary CM requirements on contractors. This does not

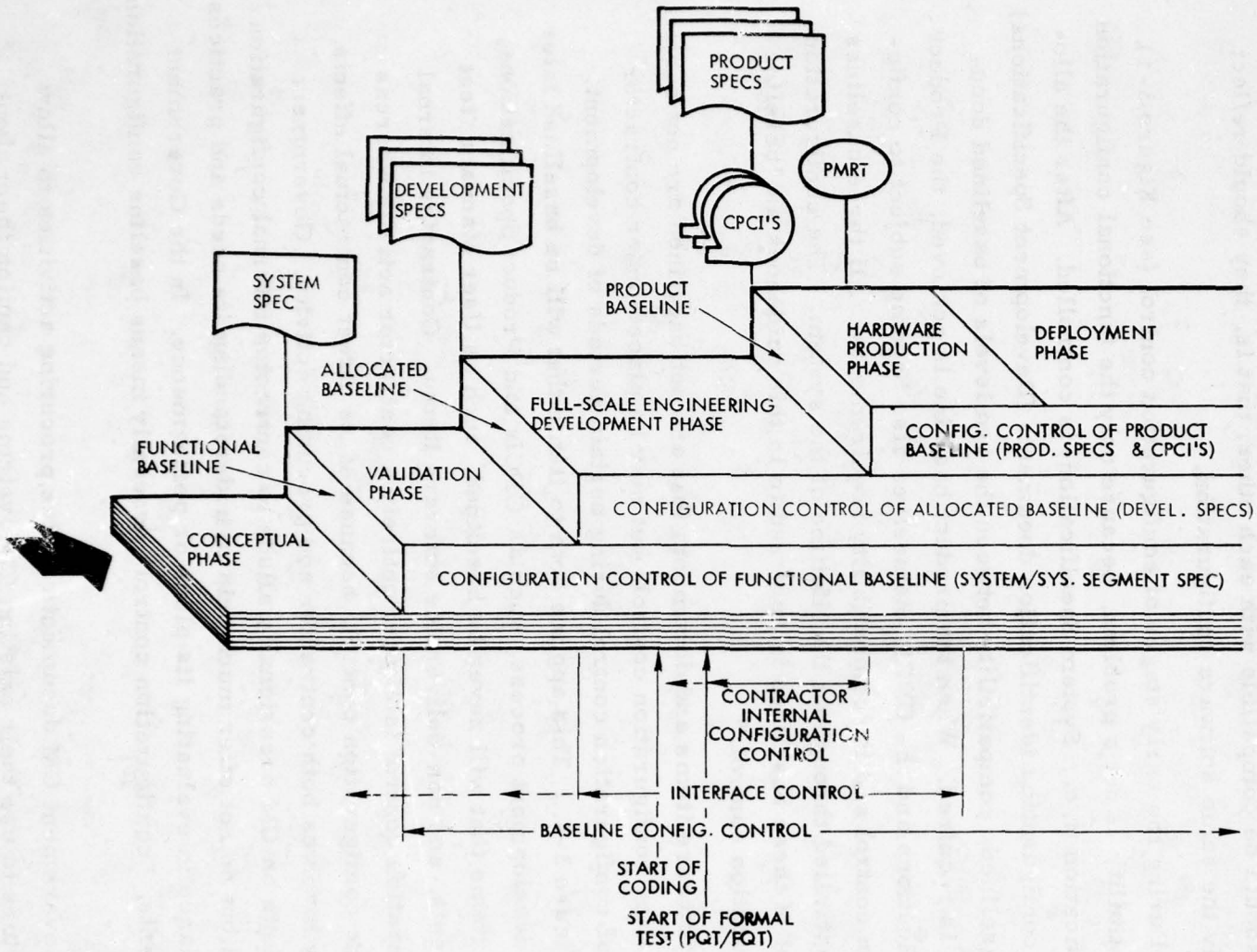



Figure 3-1. General Configuration Control Periods

relieve a procuring activity of the responsibility for evaluating contractor CM systems to ensure that they are capable of providing the visibility and control required. This responsibility applies to the contractor's methods for meeting his internal configuration control problems as well as his methods for complying with procuring activity requirements for baseline configuration control.

Interface control (also shown in Figure 3-1) is a third level of configuration control that must be considered by program offices. Formal interface control is required when two or more contractors or Government agencies are developing items whose configurations may affect each other. A separate review board (Interface Control Working Group, or ICWG), separate documentation (Interface Control Drawings or Interface Design Specifications), and separate control procedures are required for interface control, but a close working relationship is maintained with the CM groups of participating organizations.

### 3.1.3 Basic Concepts of Configuration Status Accounting

As a result of the two CM areas discussed in the preceding pages, an item's configuration is identified and changes to that configuration are proposed, evaluated, and implemented. Keeping track of the configuration identification and its changes and reporting this information are the functions of configuration status accounting.

Two major types of documents are produced by status accounting:

a) Configuration Index. This index defines the current approved configuration of an item in terms of its elements or identification documents and its approved changes.

b) Change Status Reports. These reports give the implementation status of changes to a configuration item.

These status accounting documents provide implementing personnel and management with visibility and traceability of baseline configurations and their changes. They give implementing personnel the means to coordinate the many actions that must be performed in support of changes, and they give management the means to determine if change decisions are being implemented as directed.

During development, status accounting documents are an important information source for the procuring activity and in the production and deployment phases, they are an essential means of coordinating maintenance and modification tasks that may involve many organizations in widely scattered locations.

### 3.1.4 Basic Concepts of Configuration Audits

A series of configuration-oriented audits is conducted near the conclusion of a development program to verify that the completed product satisfies its contract requirements. These audits and their characteristics are as follows:

a) Functional Configuration Audit (FCA). The CPCI FCA verifies through review of PQT and FQT test data that a CPCI's actual performance complies with the performance requirements of its Development Specifications. The FCA usually is a preparatory audit leading to scheduling of the PCA, but can be combined with the PCA. System FCAs also are conducted, usually following system testing.

b) Physical Configuration Audit (PCA). The CPCI PCA is a formal audit to verify a CPCI's product baseline documentation. The Product Specification is checked against the CPCI listing to ensure the document reflects the actual configuration, and CM records are examined to ensure that approved changes have been incorporated. The Computer Programming Manual, Users Manual, and any other documents not previously accepted also are examined. Successful completion of the PCA results in authentication of the Product Specification and establishment of the product baseline for the CPCI. When final testing of a CPCI must be done at the system level, the PCA may not occur until the production phase.

c) Formal Qualification Review (FQR). The CPCI FQR is considered a configuration audit if it is included in the Configuration Item Development Record, which is part of the CPCI Configuration Index. Representatives of AFSC and other appropriate commands review the Development Specification and PQT and FQT test data and also may witness tests or demonstrations to certify that the CPCI is qualified for its intended application. These Government participants verify that all tests required by Section 4 of the Development Specification have been accomplished and that the resulting test data proves that the CPCI satisfies the performance requirements of Section 3 of that specification. An FQR is combined with the FCA (i.e., prior to the PCA) if the required test data is available at that time. When FCA and FQR are not combined, continuity between them should be maintained to avoid duplication of effort.

Agendas and minutes for configuration audits are described in paragraph 3.5.3 of this guidebook. Additional information about audits is presented in the "Reviews and Audits Guidebook" of this series. The primary RSS on configuration audits are AFR 65-3 and MIL-STD-1521A.

### 3.1.5 Benefits of CM

A properly planned and implemented software CM program will provide the following benefits:

a) Assist management in achieving required CPCI performance, operational efficiency, and logistic support on schedule and at the lowest total life cycle cost.

b) Combine maximum freedom for design and development with appropriate configuration control for baseline items.

c) Provide knowledge of exact configurations of controlled application, test, and support CPCIs at all times.

d) Ensure that specifications and related technical data are adequate for both CM and program needs and will be available in verified form when needed.

e) Ensure that CPCI external interfaces are documented and controlled.

f) Provide efficient management of configuration changes in regard to their necessity, cost, timing, and implementation.

g) Provide appropriate uniformity of CM policies, procedures, documents, forms, and reports within DOD and between DOD and contractors.

### 3.1.6 Differences Between Software CM and Hardware CM

Government documents on configuration management generally present a hardware-oriented point of view, with only occasional consideration of the special needs of software CM. This is true even of the AFR 800 series of computer resource documents.

The differences between hardware CM and software CM arise, of course, because of the vastly different natures of the hardware and software configuration items themselves. For CM purposes, an item of hardware is an assembly of complex parts (electronic and mechanical) that have different physical characteristics (materials, dimensions, etc.) and must relate to each other precisely in complex ways. Once

built, hardware is difficult to change. For CM purposes, a computer program is an assembly of simple parts (characters or bits) in a one-dimensional, or serial, arrangement. Any physical characteristics associated with a computer program apply only to its storage medium (punched cards, magnetic tape, etc.) and storage format (bit density, tracks, etc.). A coded computer program can be changed fairly quickly and easily.

Despite these differences, documentation requirements for hardware and software items are much the same. The same general types of configuration identification specifications are used during development, and the test and support documents are similar. Documentation of the as-built configurations is very different, however. Large quantities of drawings and specs are needed to define the elements and characteristics of hardware, whereas a computer program configuration is completely and exactly represented by a computer-generated source listing. A Computer Program Product Specification is still required to make the listing meaningful, however.

A production phase is not required for computer programs because they can be duplicated quickly and inexpensively. This duplication is independent of the program's configuration itself, and therefore can be performed without reference to the Product Specification or listing. Quality control in the hardware production sense is not required for duplicated computer programs, but some verification of the accuracy of each copy should be performed through use of bit-by-bit comparison programs, checksum programs, visual inspection of listings, or other means.

Maintenance of software means correction of design defects or code errors or modification to meet changing requirements. The volume of changes required in some types of delivered programs tends to be high. One reason is that system designers depend on software's flexibility to accommodate most system changes. Another reason is that some errors in large and complex programs are not readily discovered by present testing techniques. Once software changes are designed, coded, and tested, however, the mechanics of installing them are routine.

A computer program will always give the same results if all operating conditions are exactly the same. It will not wear out or otherwise deteriorate with use, time, or environmental circumstances (although its deck, tape, or other storage medium may suffer from these things). Therefore, reliability in the sense of useful life has no meaning for software and spare parts provisioning is not a logistics concern (except for extra cards, tapes, etc).

Because of such differences between hardware and software, the standard hardware CM practices require some modification before they can be applied to software CM programs. The basic concepts are the same, however, so it is possible for system CM programs to use similar approaches in their software and hardware CM activities.

## 3.2 LIFE CYCLE VARIATIONS IN SOFTWARE CM

Configuration management is required in varying degrees throughout the entire system acquisition life cycle, generally beginning at a low level and increasing in stages as the product evolves. The major tasks required to implement CM are described in this subsection against the background of the system acquisition life cycle and the computer program life cycle.

The computer program life cycle is an independent series of activities than can be performed as part of a system acquisition life cycle or completely outside the sphere of a system acquisition. When it is part of a system acquisition, its general relationship to the system acquisition life cycle is shown in Figure 3-2. The bottom block in this figure shows the CPCI and system test periods.

The sequence of computer program development activities and related events (PDR, CDR, etc) in a computer program life cycle must occur at least once for every CPCI during a system acquisition life cycle. In addition, the basic elements of this sequence — analysis, design, coding, checkout, and test — and some of the related events and support activities may be repeated on a smaller scale in the conceptual or validation phase or in both of these if there is a need for simulations or other exploratory computer activity. The basic elements also are repeated for every computer program problem correction or modification

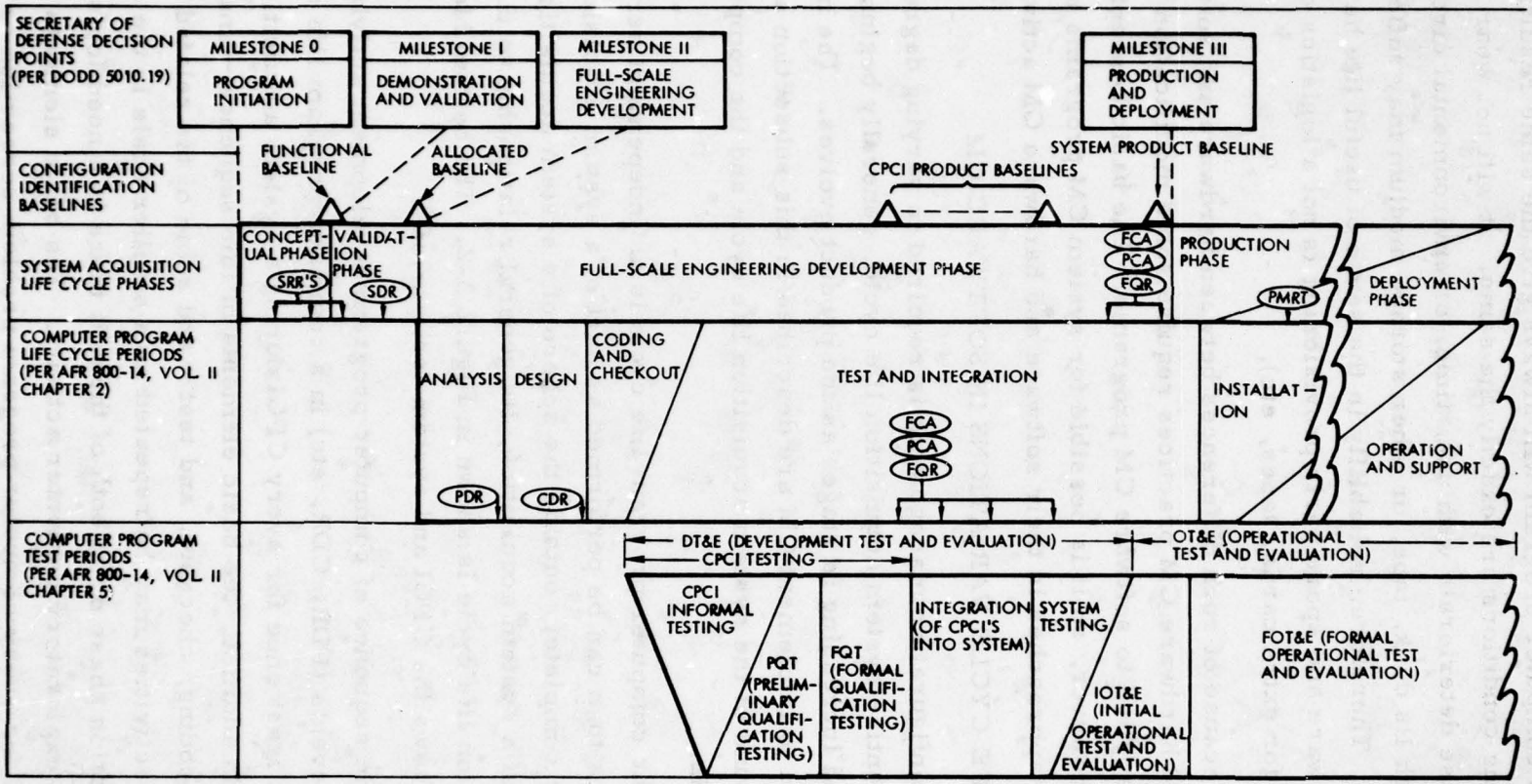



Figure 3-2. System and Computer Program Life Cycles

that is made at any point in the acquisition life cycle. During testing, this sequence may be repeated hundreds of times within a short period. Loops within the computer program life cycle also occur when it is necessary to change results of an earlier stage and then incorporate the consequences of this change in all intervening stages.

A summary of the principal CM-related tasks and milestones that occur during a representative system acquisition life cycle is presented in Figure 3-3. The general relationships shown in Figure 3-2 are assumed.

## 3.3 PROGRAM FACTORS THAT AFFECT CM REQUIREMENTS

There are many program variables that influence the planning and conduct of CM. The basic concepts described earlier are generally applicable to any development program, but the details should be tailored to the particular needs of each program.

Some of the program factors that create the greatest problems or uncertainties in applying the general concepts of CM are discussed in the following paragraphs.

### 3.3.1 Effect of Smaller Programs on Software CM

It is sometimes difficult on smaller programs to create any sense of positive CM control without burdening personnel with excessive paperwork and rules. Some of the important considerations on smaller programs are these:

a) Closer informal contact among procuring activity and contractor personnel reduces the need for formal CM procedures, forms, and approvals. However, logs or notebooks should be maintained on both sides to document key problems, decisions, and changes, and key approvals should be made on a formal basis.

b) Small programs require less configuration identification documentation. If a CPCI is being developed independently of a system, a System or System Segment Specification is not required; only a Development Specification is needed for requirements (it becomes the functional configuration identification in such cases) and a Product Specification for design. These can even be combined in a single specification. (See DID DI-E-30130, "Non-Complex Computer Program Configuration Item (CPCI) Specification." Also

## CONCEPTUAL PHASE

The purpose of the conceptual phase is to define overall mission and system requirements. The major product is the preliminary System Specification (and/or preliminary System Segment Specification).

If any software is developed for exploratory reasons during the conceptual phase, it need not be subject to CM. If some CM is desired for such items, it should be limited to control of functional characteristics.

**Major software CM-related tasks and milestones of this phase usually are:**

a. Functional Configuration Identification. *The program office or conceptual phase contractor prepares the initial version of the System/System Segment Specification.*

b. Interface Control Drawings (ICDs). The *conceptual phase contractor prepares* ICDs for system/segment interfaces with other systems/segments.

c. Configuration Control Board (CCB). The program office establishes the program CCB.

d. Interface Control Working Group (ICWG). The program office establishes an ICWG if such a group is appropriate at this stage.

e. System Requirements Review (SRR). One or more SRRs may be conducted.

f. Functional Baseline. If possible, the program office approves the System/System Segment Specification during this phase to establish the functional baseline.

## VALIDATION PHASE

If the acquisition strategy at the start of the validation phase is to concentrate on a single system, the purpose of this phase will be to validate system concepts and refine the basic characteristics of the system. Major products will be an updated System/System Segment Specification, preliminary Development Specifications, and the initial Computer Resources Integrated Support Plan (CRISP). Major software CM tasks will be as follows:

a. Functional Baseline Configuration Control. The program office places the System/System Segment Specification under baseline configuration control.

b. Status Accounting. *The program office* begins formal configuration status accounting for the System/System Segment Specification.

c. CPCI Identification. *The program office or validation phase contractor makes a preliminary identification (i.e., selection) of system CPCIs.*

d. System Specification Update. The program office or validation phase contractor updates the System/System Segment Specification.

e. Allocated Configuration Identification. The program office or validation phase contractor prepares the initial Computer Program Development Specifications.

f. Computer Resources Working Group (CRWG). The program office organizes a CRWG consisting of representatives from the implementing, using, and supporting commands.

g. System Requirements Reviews (SRRs). One or more SRRs may be conducted.

h. System Design Review (SDR). An SDR is conducted.

i. Allocated Baseline. Following SDR (System Design Review), the program office may choose to authenticate the Development Specifications to establish the allocated baseline.

Alternatives to the single-system concept for the validation phase are (1) investigation of several systems, (2) investigation of alternative subsystems only, with no system-level activities, and (3) moving directly to the full-scale development phase. Each of these alternatives would result in some variation in the above list of major products and tasks.

## FULL-SCALE ENGINEERI

### ANALYSIS

### DESIGN

The purpose of the full-scale engineering development phase (FSD) is to design, build, and te system; and to test the system under nearly operational conditions as possible. Major products

CPCIs are subject to CM during this phase. In addition to formal configuration control of ba CPCIs, preliminary Product Specifications, and Test Procedures during PQT, FQT and any oth

*Source selection may take place either at the beginning of this phase or at the end of the vali*

a. RFP CM Requirements. The program office prepares CM requirements for inclusion in the phase Request for Proposal (RFP).

b. CM Plan. Competing contractors submit abbreviated CM Plans with their proposals.

c. Contract CM Requirements. The program office prepares CM requirements for inclusion in

CM-related activities and milestones during the four FSD periods are described in the followi

The purpose of the analysis activity is to define the CPCIs in terms of functions, external and internal interfaces, storage allocation, operating sequences and data base design. The major products are preliminary Computer Program Product Specifications.

Baseline specifications are controlled during this period.

Major software CM-related tasks and milestones are:

a. Allocated Baseline Configuration Control. After authenticating the Computer Program Development Specifications (either during the validation phase or in this analysis period), the program office places them under formal configuration control.

b. Status Accounting. The program office and contractor begin formal configuration status accounting for the Development Specification.

c. Status Accounting Reports. The contractor begins to issue the Configuration Index and Change Status Reports.

d. Completed CM Plan. The contractor completes and updates his CM Plan.

e. Interface Agreements. The program office prepares interface agreements between all participating contractors.

f. Interface Control Working Group (ICWG). The program office establishes an ICWG if one is warranted and has not been established earlier.

g. Development Specification Updates. The contractor updates the Development Specifications by means of ECPs and SCNs.

h. Product Configuration Identification. The contractor prepares the initial Computer Program Product Specifications.

i. Preliminary Design Reviews. The program office and contractor conduct the CPCI Preliminary Design Reviews (PDRs).

The purpose of design is to define the CPCI structure, interface logic, and data base in sufficient detail to permit coding to take place. The major products are expanded Product Specifications.

Baseline specifications continue to be controlled during this period, but no new items are placed under control.

Major software CM-related tasks and milestones are:

a. Item Identifiers. The program office obtains Computer Program Identification Numbers (CPINs) from the AFLC Air Logistics Center (ALC) for all CPCIs and related documents, or assigns other suitable item identifiers.

b. Product Specification Updates. The contractor refines and expands the Product Specifications.

c. Critical Design Reviews. The program office and contractor conduct the CPCI Critical Design Reviews (CDRs).

d. CM Procedures. The contractor prepares detailed procedures for internal configuration control of CPCIs and Product Specifications during test and integration.

e. Software Development Library Preparation. The contractor prepares facilities and procedures for generation, control, and update of master software files and for storage and control of associated decks, tapes, discs, listings, etc.

## FULL-SCALE ENGINEERING DEVELOPMENT PHASE

### DESIGN

ment phase (FSD) is to design, build, and test system CPCIs and hardware CIs; to integrate the CPCIs and hardware CIs into the complete
lanal conditions as possible. Major products are the CRISP, CPCIs, hardware CIs, Product Specifications, and test and support documents.

dition to formal configuration control of baseline items, the contractor usually is required to implement internal configuration control for
st Procedures uring PQT, FQT and any other formal testing required prior to the CPCI product baselines.

inning of this phase or at the end of the validation phase. CM tasks associated with source selection are:

repares CM requirements for inclusion in the Statement of Work (SOW) and Contract Data Requirements List (CDRL) of the development

eviated CM Plans with their proposals.

ice prepares CM requirements for inclusion in the SOW and CDRL of the development phase contract.

our FSD periods are described in the following columns.

urpose of design is to define the CPCI
ure, interface logic, and data base in
ent detail to permit coding to take place.
ajor products are expanded Product
fications.

he specifications continue to be con-
d during this period, but no new items
ced under control.

software CM-related tasks and milestones

em Identifiers. The program office
btains Computer Program Identification
lumbers (CPINs) from the AFLC Air
ogistics Center (ALC) for all CPCIs and
lated documents, or assigns other suit-
ble item identifiers.

oduct Specification Updates. The con-
ctor refines and expands the Product
ecifications.

ritical Design Reviews. The program
ffice and contractor conduct the CPCI
ritical Design Reviews (CDRs).

M Procedures. The contractor prepares
tailed procedures for internal configura-
on control of CPCIs and Product Specifi-
tions during test and integration.

ftware Development Library Preparation.
e contractor prepares facilities and pro-
dures for generation, control, and update
master software files and for storage and
ntrol of associated decks, tapes, discs,
tings, etc.

### CODING AND CHECKOUT

The purpose of coding and checkout is to code routines and data files, to debug them (so they will compile), and to check them out (so they will produce correct results from predefined inputs).

Baseline specifications continue to be controlled. The only new items placed under control are routines that must be released early to the development library for use by other developers.

The major software CM-related task is to impose internal configuration control on all routines formally released during this period.

### TEST AND INTEGRATION

The purpose of test and integration is to test CPCIs against the requirements of their Development Specifications, to integrate CPCIs with other elements of the system, and to test the complete system against the requirements of the System/System Segment Specification.

CM activity reaches it peak during this period. During Formal Qualification Testing (FQT) and integration, many software problems usually are found, requiring changes to the computer programs, Product Specifications, Development Specifications, and usually, System/System Segment Specification.

Major software CM-related tasks and milestones are:

a. Internal Configuration Control. The contractor imposes internal configuration control on selected CPCIs and Product Specifications during Preliminary Qualification Testing (PQT) and on all CPCIs and Product Specifications during FQT. Test Procedures also are controlled.

b. Internal Status Accounting. The contractor implements his internal status accounting procedures during PQT.

c. OS/CMP. The program office participates with the using and supporting commands in preparation of the Operation/Support CM Procedures (O/S CMP).

d. Final Product Specifications. The contractor prepares the as-coded versions of the CPCI Product Specifications.

e. Configuration Audits and FQR. The program office and contractor conduct CPCI Functional Configuration Audits (FCAs), Physical Configuration Audits (PCAs), and Formal Qualification Reviews (FQRs). Testing requirements may cause some of these to be deferred to the production phase.

f. Product Baseline. Following successful PCAs, the program office authenticates the Product Specifications to establish the CPCI Product Baselines.

g. CPCI Delivery. Following successful FCAs, PCAs, and FQRs, the contractor delivers custody of the CPCIs to the program office.

h. Product Baseline Configuration Control. The program office places the Product Specifications and CPCIs under baseline configuration control.

## PRODUCTION PHASE

The purpose of the production phase is to produce complete systems (i.e., hardware and software) and deliver them to the using commands. Software activity in this phase usually consists only of the duplication of the software items required in the system. This phase begins with production approval and ends when the last complete system is delivered and accepted. Since deployment may begin early in the production phase, segments of the production and deployment phases may overlap.

The production phase contains two key transition points in system acquisition: system turnover from AFSC to the using command and Program Management Responsibility Transfer (PMRT) from AFSC to the supporting command. At system turnover, AFSC relinquishes all system management responsibilities and at PMRT, all system support responsibilities including CM, except for identified residual tasks and phase-out responsibilities. A software support group belonging to AFLC or another contractor usually assumes the development contractor's programming support responsibilities after PMRT. Major CM-related task and milestones during the production phase are:

a. Status Accounting. All participating commands and contractors begin system-level status accounting procedures, and the status accounting agency or contractor begins to issue the Configuration Identification Index (CII) and Configuration Status Accounting Reports (CSARs).

b. Transfer of CM Responsibility. At PMRT, the program office transfers system CM responsibilities to the supporting command.

c. O/S CMP. After PMRT, all system participants comply with the CM requirements documented in the Operation/Support CM Procedures (O/S CMP).

d. Time Compliance Technical Orders. After PMRT, AFSC and the supporting command begin using Time Compliance Technical Orders (TCTOs) for CPCI changes.

e. Residual CM Tasks. After PMRT, the program office completes identified residual CM tasks before phasing out.

## INSTALLATION

The purpose of deployment is t
integrate, check out, and dem
Deployment begins with deliv
when the system is removed fr

The deployment phase consists

The purpose of installation is to p
tem for operational use. It inclu
and running computer programs,
puter programs to different sites,
the computer programs to verify th
ate with the required level of co
support of the total system in the
environment.

Software and related documents
CM in accordance with the O/S
is responsible for CM at operation
the using command wishes to be r
change control of computer progr
data required for the direct perfo
mission. When security, safety,
restrictions are involved, special
control procedures are required th
involve complex checks and bala
mandatory security keys and acce

| TEST AND INTEGRATION | PRODUCTION PHASE | DEPLOYMENT PHASE | |
|---|---|---|---|
| | | INSTALLATION | OPERATION/SUPPORT |
| hardware CIs into the complete , and test and support documents.<br><br>ternal configuration control for<br><br>ist (CDRL) of the development<br><br>rpose of test and integration is to test against the requirements of their Devel- Specifications, to integrate CPCIs with elements of the system, and to test the te system against the requirements of the /System Segment Specification.<br><br>tivity reaches it peak during this period. Formal Qualification Testing (FQT) and ation, many software problems usually are requiring changes to the computer pro- Product Specifications, Development fications, and usually, System/System nt Specification.<br><br>software CM-related tasks and milestones<br><br>ternal Configuration Control. The con- actor imposes internal configuration con- ol on selected CPCIs and Product Specifi- ations during Preliminary Qualification sting (PQT) and on all CPCIs and Product ecifications during FQT. Test Procedures lso are controlled.<br><br>ternal Status Accounting. The contractor plements his internal status accounting ocedures during PQT.<br><br>S/CMP. The program office partici- tes with the using and supporting mmands in preparation of the Opera- ion/Support CM Procedures (O/S CMP).<br><br>nal Product Specifications. The con- actor prepares the as-coded versions of e CPCI Product Specifications.<br><br>nfiguration Audits and FQR. The pro- am office and contractor conduct CPCI nctional Configuration Audits (FCAs), ysical Configuration Audits (PCAs), and rmal Qualification Reviews (FQRs). sting requirements may cause some of se to be deferred to the production hase.<br><br>oduct Baseline. Following successful CAs, the program office authenticates the oduct Specifications to establish the CPCI oduct Baselines.<br><br>PCI Delivery. Following successful FCAs, CAs, and FQRs, the contractor delivers stody of the CPCIs to the program office.<br><br>oduct Baseline Configuration Control. e program office places the Product ecifications and CPCIs under baseline nfiguration control. | The purpose of the production phase is to produce complete systems (i.e., hardware and software) and deliver them to the using commands. Software activity in this phase usually consists only of the duplication of the software items required in the system. This phase begins with production approval and ends when the last complete system is delivered and accepted. Since deployment may begin early in the production phase, segments of the production and deployment phases may overlap.<br><br>The production phase contains two key transition points in system acquisition: system turnover from AFSC to the using command and Program Management Responsibility Transfer (PMRT) from AFSC to the supporting command. At system turnover, AFSC relinquishes all system management responsibilities and at PMRT, all system support responsibilities including CM, except for identified residual tasks and phase-out responsibilities. A software support group belonging to AFLC or another contractor usually assumes the development contractor's programming support responsibilities after PMRT. Major CM-related task and milestones during the production phase are:<br><br>a. Status Accounting. All participating commands and contractors begin system-level status accounting procedures, and the status accounting agency or contractor begins to issue the Configuration Identification Index (CII) and Configuration Status Accounting Reports (CSARs).<br><br>b. Transfer of CM Responsibility. At PMRT, the program office transfers system CM responsibilities to the supporting command.<br><br>c. O/S CMP. After PMRT, all system participants comply with the CM requirements documented in the Operation/Support CM Procedures (O/S CMP).<br><br>d. Time Compliance Technical Orders. After PMRT, AFSC and the supporting command begin using Time Compliance Technical Orders (TCTOs) for CPCI changes.<br><br>e. Residual CM Tasks. After PMRT, the program office completes identified residual CM tasks before phasing out. | The purpose of deployment is to transport system elements to operational sites; to install, integrate, check out, and demonstrate the system; and to operate and support the system. Deployment begins with delivery and acceptance of the first operational unit and ends when the system is removed from the operational inventory.<br><br>The deployment phase consists of two segments: (a) installation and (b) operation and support.<br><br>The purpose of installation is to prepare a system for operational use. It includes loading and running computer programs, adapting computer programs to different sites, and checking the computer programs to verify that they operate with the required level of confidence in support of the total system in the operational environment.<br><br>Software and related documents are subject to CM in accordance with the O/S CMP. AFLC is responsible for CM at operational sites unless the using command wishes to be responsible for change control of computer programs or mission data required for the direct performance of its mission. When security, safety, or nuclear restrictions are involved, special configuration control procedures are required that may involve complex checks and balances with mandatory security keys and access codes. | The purpose of operation and support is to ensure that a system is able to accomplish mission objectives.<br><br>This period usually requires modification of the original software configuration and its documentation to correct errors, improve performance, adapt to changes in hardware or in system requirements, or incorporate improvements based on operational use.<br><br>Software and related documents are subject to CM in accordance with the O/S CMP. AFLC is responsible for CM at operational sites except for areas in which the using command wishes to be responsible, as mentioned in the preceding column. |

**Figure 3-3. Software CM During Life Cycle Periods**

see subsection 3.3.5, "Tradeoff Factors for Documentation Levels" in SAE Guidebook for Computer Program Documentation Requirements.)

c) Some kind of CM plan, even a brief one, should be prepared. The eight-step CM planning process outlined in paragraph 3.5.1 of this guidebook is as valid for small programs as for any others.

### 3.3.2 Effect of Multiple Versions or Locations on Software CM

Programs with multiple versions of the same CPCI or with the same CPCI at different operational locations have special CM problems and must employ careful planning and coordination to maintain the software successfully. Some of the possible situations are as follows:

a) Different Locations, Central Control. When a single CPCI configuration is required for several operational locations, the configuration usually is controlled from one location. The controlling center receives information from the other locations about problems and proposed corrections, coordinates the information with the procuring activity for analysis and approval, and releases approved changes to all locations. (AFSCP 800-7, paragraph 6-6h(3), describes use of Version Description Documents (VDDs) for multiple-location systems.)

b) Different Locations, Local Control. Sometimes copies of the same CPCI configuration are subject to local control at each of a series of locations. Each location may be required to submit information on problems and changes to the other locations for investigation of possible impacts, but usually no attempt is made to keep the configurations identical. This situation is not feasible, of course, if gradual divergence of CPCI configurations is likely to affect system operation or maintenance.

c) Multiple Versions of Single CPCI. A single CPCI may have more than one version under development at the same time. Possible instances include the following:

1) Follow-on development to a contract currently in its operation and support phase.

2) A CPCI developed in two or more phases, each independently operational, such as initial operating capability and full operating capability. This approach, sometimes called loop development, may be done to utilize manpower at a steady rate.

3) An addition to a product currently in development. For example, an ECP is issued by the procuring activity to expand capability of a CPCI currently at the product baseline, with the new version to be started at the allocated baseline.

In each of these cases, separate Development and Product Specifications normally are required for each separate version of a CPCI. Alternate documentation approaches are to treat each version as a different type and document it in the same set of specifications or to prepare Addendum Specifications to the basic specifications. (See Subsection 4.1.5, "Special Problems: Multiple Locations and Modified CPCIs" in this guidebook.) A major problem in multiple-version control is the coordination of changes among the versions. For example, loop A, currently in system test, has system, allocated, and product baselines and is subject to baseline configuration control, while loop B, in FQT, has only system and allocated baselines and is subject only to internal configuration control. A change of small consequence to loop B might have severe implications for loop A. Therefore, all versions must be considered whenever a change is incorporated in any of them.

CM responsibilities for multiple versions or multiple locations should be documented in the CRISP and detailed CM procedures for these situations should be documented in the O/S CMP.

### 3.3.3 Effect of Advanced Software Technology on Software CM

A 1974 study* indicated that software CM procedures require only slight modification to accommodate structured programming, top-down programming, and other newer software technologies. The major effects on CM are these:

a) Detailed "as-built" flow charts are unnecessary for portions of a computer program for which structured source code listings are available, and considerable cost savings should be possible if they are eliminated as a requirement. (Detailed "build-to" flow charts are still required for Critical Design Reviews, however.) Another

*Boehm, B. W., et al., "The Impact of New Technologies on Software Configuration Management," TRW Systems Group, Redondo Beach, California, 10 June 1974.

1974 study* defined the changes that are necessary in the MIL-STD-483 requirements for a Computer Program Product Specification to permit such substitution of structured source listings for detailed flow charts.

b) Under top-down development, some programs become part of an integrated, operable entity shortly after coding starts and require contractor internal configuration control then instead of at PQT or FQT.

### 3.3.4 Effect of Firmware, Microprocessors, Etc. on Software CM

AFSC policy on management of firmware, microprocessors, and other recent developments in computer hardware technology are defined in AFSC Supplement 1 to AFR 800-14, Volume I:

a) Computer firmware will be managed as CPCIs or components and documented accordingly. (Paragraph 3m(8).)

b) Firmware development equipment, Read Only Memory (ROM) programming equipment, and ROMs will be managed as equipment CIs or components and documented accordingly. (Paragraph 3m(8).)

c) Computer programs will be developed and managed as CPCIs without regard to the physical characteristics of the ultimate storage medium (e.g., magnetic tape or core, ROM, or programmable ROM). (Paragraph 3m(8).)

d) Microprocessors and microcomputers will be considered microelectronics/circuits for parts selection and control. (Paragraph 3e(2).)

## 3.4 CM RESPONSIBILITIES

### 3.4.1 Government CM Responsibilities

#### 3.4.1.1 Command Responsibilities

General CM responsibilities of Air Force commands participating in software acquisition and operation are as follows:

a) Implementing Command. This is the command responsible for acquisition of a system. It is responsible for development, operation, and maintenance, including associated CM tasks, throughout development, development testing, and a

*Ortega, L. H., "Documentation Standards, Final Report," Vol. VII of Structured Programming Series, IBM, Federal Systems Division, Gaithersburg, Maryland, 21 Sept. 1974.

portion of operational testing. The implementing command transfers program management responsibility (usually including CM responsibility) to the supporting command at PMRT (Program Management Responsibility Transfer) and turns over system operation and maintenance to the using command sometime prior to PMRT. For purposes of this guidebook, AFSC is the implementing command.

b) Supporting Command. This command provides CM, logistic support, and other kinds of direct support required by a system during deployment and operational use. The supporting command participates with the procuring activity in program planning and monitors aspects of development related to its support responsibilities. AFLC is the supporting command for most AFSC acquisition programs.

c) Using Command. This command is primarily responsible for operational use and maintenance of a system. It may rely on the supporting command for computer program configuration management or may perform this task itself, as in the case of computer programs required for the direct performance of its mission. The using command participates with the procuring activity and supporting command in program planning and remains involved in program activities during development, test, production, and deployment.

3.4.1.2 Program Office Responsibilities

A Program Office (PO) is an Air Force procuring activity established within a product division (ASD, SAMSO, etc.) early in the validation phase for acquisition of a new system. The PO is headed by a Program Manager (PM) who is responsible for overall program management. He is supported by groups of functional specialists in the various disciplines required, including configuration management, who may be assigned directly to the PO or may be members of the Product Division functional staff. The charter of the PO is defined in a HQ USAF Program Management Directive (PMD) or in a directive from the AFSC Commander or Vice Commander or the Product Division Commander. (The rest of this guidebook assumes a PMD as the guiding directive for the PO.)

Prior to formation of a PO, a Program Cadre is organized early in the conceptual phase to perform initial planning and studies for system acquisition and to prepare initial versions of the Program Management Plan (PMP) and other planning documents that form the basis for program approval and entry into the validation phase.

Program Manager CM Responsibilities. The Program Manager is responsible for establishing and implementing a CM program based on AFR 65-3 (including Appendix F) that will identify, document, and control the functional and physical characteristics of all CPCIs under development. His primary planning document is the Program Management Plan (PMP), whose preparation and maintenance he directs. He must ensure that the CM needs of AFLC and the using command are incorporated into the PMP, supporting plans, and other system documents prepared by the PO.

The Program Manager also establishes and chairs the system-level Configuration Control Board (CCB). (See subsection 5.2.3.1 for CCB details.)

Configuration Management Office (CMO) Responsibilities. The Configuration Management Office (CMO) in the Program Office is responsible for establishing and implementing the detailed policies and procedures for software CM under the Program Manager's direction. Specific responsibilities of the CMO are likely to include the following:

a) Coordinates CM requirements with using and supporting agencies.

b) Reviews contractor CM Plans.

c) Audits contractor implementation of CM Plans.

d) Ensures that functional, allocated, and product configuration identifications of system CPCIs are documented.

e) Acts as the focal point within the PO for centralized specification control and CPCI control.

f) Receives, reviews, processes, and distributes ECPs to the affected agencies and contractors.

g) Controls engineering changes to documents and CPCIs.

h) Provides the secretariat for the Program Office CCB.

i) Ensures that system level configuration status accounting records are maintained and reports are prepared and distributed.

j) Plans and conducts configuration audits jointly with the affected contractors.

k) Prepares PMRT package for transfer to AFLC.

### 3.4.2 Contractor CM Responsibilities

Contractor CM responsibilities must be tailored to each contract. They will vary a great deal, depending on the size and complexity of the development task and on other factors. (Program factors that affect CM requirements are discussed in 3.3.) Following is a list of basic CM tasks that often are imposed on software development contractors:

a) CM Plan. Prepare a CM Plan that documents contractor responsibilities and procedures for implementing CM. Usually prepared as part of the proposal in response to a task in the Instructions for Preparation of Proposal (IFPP), and then updated in the early part of the contract period in response to a task in the SOW and an entry in the CDRL.

b) Configuration Identification and Baselines.

1) System Specification/Functional Baseline. Participate in completion of functional baseline documentation for the system by providing updates to the System Specification.

2) Development Specification/Allocated Baseline. Define the allocated baseline for the CPCIs and document the baseline in Computer Program Development Specifications. (Or, if such specifications were previously prepared and are part of the contract requirements, prepare and submit changes or revisions.)

3) Product Specification/Product Baseline. Define the product baseline for the CPCIs and document the baseline in Computer Program Product Specifications. If any previously unidentified computer program Government inventory items are to be used on the system, identify and document them.

c) Identification Numbering. Assign unique identification numbers to all CPCIs and controlled components of CPCIs and use these numbers to identify code and recording media.

d) Configuration Control.

1) Specification Maintenance. Maintain the configuration identification specifications.

2) Baseline Configuration Control. Propose and accomplish changes to baseline identification specifications.

3) Contractor Internal Configuration Control. Establish an internal configuration control system for non-baselined documents and computer programs during CPCI qualification testing.

4) Interface Control. Establish interface control working relationships with other participating contractors. (On a large system, one contractor may be given responsibility for establishing and controlling all physical and functional interfaces between CPCIs, including CPCIs that are the responsibility of other contractors.)

e) Status Accounting. Maintain configuration status accounting records and issue status accounting reports at regular intervals.

f) Configuration Audits. Prepare for and conduct jointly with the program office an FCA and a PCA for each new or modified CPCI. Assist the program office in conducting FQRs.

g) Subcontractors/Vendors. Require subcontractors and vendors to plan and implement appropriate CM measures, and monitor the implementation.

This set of contractor CM tasks is derived from Attachments 3 and 5 of AFSCP 800-7. Attachment 3 discusses tailoring of CM requirements in SOWs and proposals, and Attachment 5 contains examples of contractual clauses for CM requirements.

## 3.5 PLANNING SOFTWARE CM PROGRAM REQUIREMENTS

### 3.5.1 CM Planning Process

The principal elements of a software CM program must be identified in some methodical way. The following sequence of eight steps shows the basic logic involved:

a) Primary Items. Identify the specific items to be developed: CPCIs, specifications, user manuals, etc. Basic source of this information: the PMP (Program Management Plan) and Section 4 of this guidebook.

b) Primary Tasks. Identify the tasks and performers required to develop the primary items: analysis of CPCI requirements by contractor, design of CPCI by contractor, coding of CPCI by contractor, etc. Basic source of this information: the PMP.

c) Primary Milestones. Identify the milestone events that will mark accomplishment of the primary tasks: SRRs, SDR, PDRs, CDRs, FQR, PMRT, etc. Basic source of information: the PMP.

d) Periods of Control. Define the period of configuration control for each primary item. Basic guidance: the PMP and applicable RSS (regulations, specifications, and standards).

e) Types of Control. Identify the type or types of configuration control required for each primary item during its control period: baseline configuration control, contractor internal configuration control, and/or interface control. Basic guidance: the PMP, applicable RSS, and Section 5 of this guidebook.

f) Secondary Items. Identify the deliverable CM-related data items that will be required to implement CM for the primary items: CPDP, CMP, CRISD, ECPs, SCNs, etc. Basic guidance: applicable RSS and paragraph 3.5.3 of this guidebook.

g) Secondary Tasks. Identify the tasks and performers required to produce and maintain the secondary items: contractor prepares and updates CPDP, CMP, and CRISD, etc. Basic guidance: applicable RSS and paragraph 3.5.3 of this guidebook.

h) Secondary Milestones. Identify the milestone events that will measure accomplishment of the secondary tasks: due dates for CPDP, CMP, CRISD, and other secondary items; functional, allocated, and product baseline points; FCAs and PCAs, etc. Basic guidance: applicable RSS and paragraph 3.5.3 of this guidebook.

Note that the first three steps identify the acquisition program products and other elements that establish CM needs and background conditions. The next two steps define configuration control parameters, and the last three identify the elements required to implement a CM program. These results establish the basic features of the entire CM program and provide most of the CM information needed in the planning documents described in the next subsection.

### 3.5.2 CM Planning Documents

The foundations of a CM program for computer resources acquisition are defined in a series of four documents produced initially by the Program Office during the conceptual or validation phase:

a) Program Management Plan (PMP)

b) Computer Resources Integrated Support Plan (CRISP)

c) Statement of Work (SOW)

d) Contract Data Requirements List (CDRL)

The PMP and CRISP are maintained by the Program Office until turnover of equipment and transfer of program management responsibility. The CRISP continues to be maintained thereafter, usually by the supporting command. The SOW and CDRL are prepared initially as parts of a Request for Proposal (RFP) and then become parts of the contract. Each separate procurement requires its own tailored SOW and CDRL.

These four planning documents and their CM roles are described in the following subsections. A Program Office CM Plan is an optional addition to this group that should be considered.

#### 3.5.2.1 Program Management Plan (PMP)

The characteristics of the Program Management Plan (PMP) are as follows:

a) Purpose. The PMP is the implementing command's master plan for the management of an entire system acquisition program. It describes the integrated time-phased tasks and the resources required to accomplish the acquisition task specified in the Program Management Directive (PMD). It includes the computer resource management approaches of the supporting and using commands and contractors. The PMP is directive on all participating commands.

b) Origination and Maintenance. The Program Office prepares and issues the PMP during the conceptual phase, as soon as possible after approval of the development program (unless required earlier by the PMD or AFSC Form 56) and following coordination with the supporting and using commands. The PMP should be updated regularly to reflect new guidance from higher authority and approved management planning additions and changes by Government

agencies and contractors, especially for the next phase of the acquisition. The PMP must agree with the direction of the PMD, AFSC Forms 56, and any AFSC field command supplementary direction.

c) Usage. The Program Manager uses the PMP to schedule and direct the entire system acquisition process, and all participating Government agencies and higher level decision-makers use the PMP as a management baseline for further planning activity. As a result of the PMP, a Computer Resources Working Group (CRWG) is created and a Computer Resources Integrated Support Plan (CRISP) is prepared. The CM Plan and other development plans also are prepared by direction of the PMP and, when approved, become part of the PMP, either directly or by reference.

d) CM Portion of PMP. AFR 800-14, Volume II (paragraph 3.7), calls for the inclusion of "CM concepts" in the PMP when applicable. These concepts should cover the following CM topics for the entire system, including the hardware and software portions of the computer resources:

1) Major CM milestones

2) CM Plan requirements

3) CM Organization (CMO and CCB)

4) Configuration identification

5) Configuration control (including specification maintenance and interface control)

6) Configuration status accounting (including any automated techniques required)

7) Configuration audits (FCAs and PCAs)

8) Reference to Government directives applicable to the CM program.

e) Directive and Guidance Documents.

1) AFR 800-2: Attachment 3 and AFSC Supplement 1

2) AFR 800-14, Volume I: Paragraph 3m

3) AFR 800-14, Volume II: Paragraph 3-7

4) AFSCP 800-3: Paragraphs 2-22a and b; 3-6; 3-7; Attachments 3 and 4 (especially paragraph 5b)

### 3.5.2.2 Computer Resources Integrated Support Plan (CRISP)

The characteristics of the Computer Resources Integrated Support Plan (CRISP) are as follows:

a) Purpose. The CRISP identifies the computer resources required by implementing, supporting, and using commands throughout the system life cycle and describes the plan for providing these resources at the required times. Especially important subjects covered by the CRISP are the identification and acquisition of support software, support hardware, and related documentation required for verification and validation testing, installation, operation, and support. The CRISP includes organizational relationships and responsibilities for management and technical support of the computer resources. A CRISP is mandatory for all AFSC system acquisitions involving software.

b) Origination and Maintenance. A Computer Resources Working Group (CRWG) formed early in the acquisition period is responsible for preparation and update of the CRISP. The CRWG consists of representatives from the implementing, supporting, and using commands and is chaired initially by the Program Manager. After system turnover and transfer of program management responsibility, the membership and chairmanship are changed, with the supporting command usually assuming chairmanship. The initial CRISP should be prepared before the Milestone II review to permit key provisions to be reflected in the full-scale engineering development contracts. The CRISP must be responsive to direction expressed in the PMD and PMP and should accommodate contractor recommendations (contained in the CRISD, or Computer Resources Integrated Support Data) for support software, support hardware, documentation, and other support resources. The CRISP is maintained throughout the system life cycle.

c) Usage. The CRISP is used as the basis for planning and procurement of computer resources required for system acquisition and operation. After turnover and transfer, it continues to function as the basic agreement between the supporting and using commands for management and support of computer resources. Planning details of the CRISP are incorporated into updates of the PMP master schedule and provide part of the basis for the contractor Computer Program Development Plan (CPDP).

d) CM Portion of CRISP. According to AFR 800-14, Volume II (paragraph 3-8), the CRISP should include, as applicable: "Planning for configuration management of computer programs, including the assignment of configuration control responsibilities during the deployment phase.

This planning will reflect the operational and support concepts for the system." The CM procedures for the system deployment phase are defined later in the Operational/Support Configuration Management Procedures (O/S CMP) (see Comment 2 below).

e) Directive and Guidance Documents.

1) AFSC Supplement 1 (paragraph 3m(5)) to AFR 800-14, Volume I.

2) AFR 800-14, Volume II: paragraphs 2-4b, 3-8, 3-10, and 10-3.

f) Comments.

1) AFR 800-14, Volume I, defines "computer resources" as "the totality of computer equipment, computer programs, associated documentation, contractual services, personnel and supplies." Computer data should be included in this list and probably computer facilities also.

2) The CM portion of the CRISP is expanded in the Operational/Support Configuration Management Procedures (O/S CMP). This document, written by the supporting and using commands in conjunction with the implementing command before the end of the full-scale engineering development phase, defines the CM procedures to be used during system deployment. (See AFR 800-14, Volume II, paragraph 6-10.)

3.5.2.3 Statement of Work (SOW)

The characteristics of the Statement of Work (SOW) are as follows:

a) Purpose. The SOW defines the exact scope of a contractor's task for the entire contract period. It identifies specific engineering and design tasks, program management tasks, and management support tasks (including CM) in sufficient detail to provide a firm basis of understanding between the procuring activity and the contractor. SOWs may be limited to the tasks of a single acquisition phase or may span a combination of phases, depending on the particular procurement.

b) Origination and Maintenance. The Program Office prepares an SOW initially as part of a Request for Proposal (RFP). Following contractor selection, revisions to the SOW and other parts of the contract usually are negotiated between the Program Office and the contractor. The final negotiated SOW can be modified only through a formal contract change control process.

c) Usage. When part of an RFP, the SOW is used by bidding contractors to define the basic tasks for their proposed development effort. When part of a contract, the SOW defines the actual tasks to be performed by the contractor.

d) CM Portion of SOW. The software CM requirements in the SOW should be tailored to the specific needs of the contract by someone who knows software CM well. The CM requirements should comply with the CM provisions of the PMP, should indicate clearly the degree of CM desired, and should include a tailored list of compliance specifications and standards that the contractor is to follow.

e) Directive and Guidance Documents.

1) AFR 65-3: paragraph 1-5f

2) AFR 800-14, Volume II: paragraph 8-3

3) AFSCP 800-7, Attachments 3 and 5

4) SAE Guidebook for Statements of Work and Requests for Proposal.

3.5.2.4 Contract Data Requirements List (CDRL)

The characteristics of the Contract Data Requirements List (CDRL) are as follows:

a) Purpose. A CDRL identifies all deliverable data items, including CPCIs (see Comment 1 below), that a contractor must deliver and references the Data Item Descriptions (DIDs) that define data item contents and formats. It also specifies submission dates, numbers of copies, distribution, related SOW task statements or contract provisions, inspection and acceptance criteria, and other related information. It also may reference backup sheets that tailor the instructions of a referenced DID or that contain other supplementary information.

b) Origination and Maintenance. A CDRL is prepared by the Program Office on a series of DD Forms 1423 or AFSC Forms 707, 708, or 709. Initially it is part of the RFP, then becomes part of the contract. As part of the contract, it can be modified only through a formal contract change control process.

c) Usage. When part of an RFP, the CDRL is used by bidding contractors as the basis for the documentation tasks and CPCI delivery tasks of their proposals. When part of a contract, the CDRL defines the actual data items and CPCI items to be delivered by the contractor.

d) CM Portion of CDRL. All data items needed for proper accomplishment of the CM tasks outlined in the SOW should be listed in the CDRL. The most commonly used data items for software CM are described in paragraph 3.5.3 of this guidebook. Not all of these are needed for all contracts. Each data item should be selected on the basis of actual need, and its DID should be tailored to the specific job. (The SAE Guidebook for Computer Program Documentation Requirements describes the data item selection and tailoring process.)

e) Directive and Guidance Documents.

1) AFR 310-1

2) DOD 5000.19-L, Volume II

3) SAE Guidebook for Computer Program Documentation Requirements.

f) Comments.

1) The Armed Services Procurement Regulations (ASPR) consider computer programs to be data items and require that they be listed on a CDRL along with specifications and other documents. AFSC requires that they also be listed in the contract schedule.

2) In addition to the CM Plan listed in the CDRL as a contract requirement, an abbreviated CM Plan adequate for source selection can be obtained from bidders by including a requirement for it in the "Instructions for Preparation of Proposals (IFPP)." (See AFSCP 800-7, Attachment 3, paragraph 4).

### 3.5.3 Deliverable CM Data Items

Configuration management depends on a variety of data items for establishing CPCI identification and controlling and recording changes to the CPCI identification documents. Figure 3-4 lists a set of basic data items used for these purposes and identifies the related compliance standards usually referenced in contracts. The appendices of MIL-STD-490 or MIL-STD-483 contain the compliance requirements for most of these items.

Because the configuration identification specifications and related documents listed as items 3 through 10 in box B of Figure 3-4 are discussed briefly in subsection 4.2 of this guidebook and in detail in the SAE Computer Program Requirements Guidebook, they are not discussed further here.

| A. CM PLANNING | |
|---|---|
| DID (DATA ITEM DESCRIPTION) | 490/483 APPENDICES |
| 1. DI-S-30567, DI-S-30567A, COMPUTER PROGRAM DEVELOPMENT PLAN (CPDP) | - - - |
| 2. DI-E-3108, CM PLAN (CMP) | /I |
| 3. DI-S-30569, COMPUTER RESOURCES INTEGRATED SUPPORT DATA (CRISD) | - - - |

| B. CONFIGURATION IDENTIFICATION | |
|---|---|
| DID (DATA ITEM DESCRIPTION) | 490/483 APPENDICES |
| 1. DI-E-30145, DI-A-30008, COMPUTER PROGRAMS, DATA AND PRINTOUTS | /VI |
| 2. DI-E-3121, VERSION DESCRIPTION DOCUMENT (VDD) (COMPUTER PROGRAMS) | /VIII |
| 3. DI-E-3101, SYSTEM SPEC. | I/III |
| 4. DI-E-3117, SYSTEM SEGMENT SPEC. (MODIFICATION PROGRAM) | I/III |
| 5. DI-E-3119A, DI-E-30139 COMPUTER PROGRAM DEVELOPMENT SPEC | VI/VI |
| 6. DI-E-3120A, DI-E-30140, COMPUTER PROGRAM PRODUCT SPEC | VIII/VI |
| 7. DI-E-7031, INTERFACE CONTROL DRAWING (ICD) | /II |
| 8. DI-E-30144, COMMON DATA BASE DESIGN SPEC | - - |
| 9. DI-E-3104, ADDENDUM SPEC | /IV |
| 10. DI-E-3105, INVENTORY ITEM SPEC | XII/V |

| C. CONFIGURATION CONTROL | |
|---|---|
| DID (DATA ITEM DESCRIPTION) | 490/483 APPENDICES |
| 1. DI-E-3127, ADVANCE CHANGE/ STUDY NOTICE (ACSN) | /XIV |
| 2. DI-E-3128, ENGINEERING CHANGE PROPOSALS (ECPs) | /XIV* |
| 3. DI-E-3134, SPECIFICATION CHANGE NOTICE (COMPUTER PROGRAM) | /VIII |
| 4. (NO DID EXISTS), NOTICE OF REVISION (NOR) | /XIV** |
| 5. DI-E-3129, REQUEST FOR DEVIATIONS/WAIVERS | *** |

* SUPPLEMENTS DOD-STD-480A OR MIL-STD-481A.

** SUPPLEMENTS DOD-STD-480A.

*** COMPLIANCE REQUIREMENTS ARE IN DOD-STD-480A OR MIL-STD-481A.

| D. CONFIGURATION STATUS ACCOUNTING | |
|---|---|
| DID (DATA ITEM DESCRIPTION) | 490/483 APPENDICES |
| 1. DI-E-3122, CONFIGURATION INDEX (COMPUTER PROGRAM) | /VIII |
| 2. DI-E-3123, CHANGE STATUS REPORT (COMPUTER PROGRAM) | /VIII |
| 3. DI-E-3116, SYSTEM ALLOCATION DOCUMENT (SAD) | /XI |
| 4. DI-E-3107, INSTALLATION COMPLETION NOTIFICATION | /XV |
| 5. DI-E-3133, CM ACCOUNTING REPORTS (MACHINE OR MANUALLY PREPARED) | * |

* COMPLIANCE REQUIREMENTS ARE IN MIL-STD-482A.

| E. CONFIGURATION AUDITS | |
|---|---|
| DID (DATA ITEM DESCRIPTION) | 490/483 APPENDICES |
| 1. DI-A-3029, AGENDA-DESIGN REVIEWS, CONFIGURATIONS AUDITS AND DEMONSTRATIONS | /XII* |
| 2. DI-E-3118, MINUTES OF FORMAL REVIEWS, INSPECTIONS AND AUDITS | /XII* |

* THESE DATA ITEMS ALSO ARE DISCUSSED IN MIL-STD-1521A.

Figure 3-4. CM Data Item Tree

For the remaining CM data items in the tree, the general periods of usage are shown in Figure 3-5 and their purpose, origination, maintenance, usage, and related documentation are described in Table 3-1.

The Computer Program Documentation Requirements Guidebook mentioned above also describes the procedure for contractual acquisition of data items and discusses the tailoring of Data Item Description (DIDs) to the minimum essential requirements of a contract.

## 3.6 MONITORING CONTRACTOR CM

Contractor CM activities can be monitored by a program office CMO through the contractor's own documents and reports, through the Government's in-plant representatives, through configuration audits, and through other parts of the program office. A useful tool for all of these approaches is the checklist. Following are brief descriptions of these subjects:

a) Monitoring CM Through Contractor Documents and Reports. Most of the configuration identification documentation, configuration control data, and status accounting data of a program are produced by the development contractor and provided to the program office as required CDRL items. This material is a major source of visibility into the contractor's CM performance. CMO personnel also should attend design reviews to observe contractor methods of document maintenance and design change.

b) Monitoring CM Through Contractor Administrative Services (CAS). CMOs should avail themselves of aid from Contract Administrative Services (CAS) at the contractor's development facility when appropriate skills and resources are available. ASPR 1-406, Contract Administrative Functions, assigns certain CM functions to CAS, including the following:

   1) Evaluation and monitoring of contractor CM systems.

   2) Monitoring configuration control systems.

   3) Reviewing Class I ECPs and commenting on their feasibility and need.

   4) Assisting in ECP price analysis.

   5) Reviewing Class II changes for proper classification.

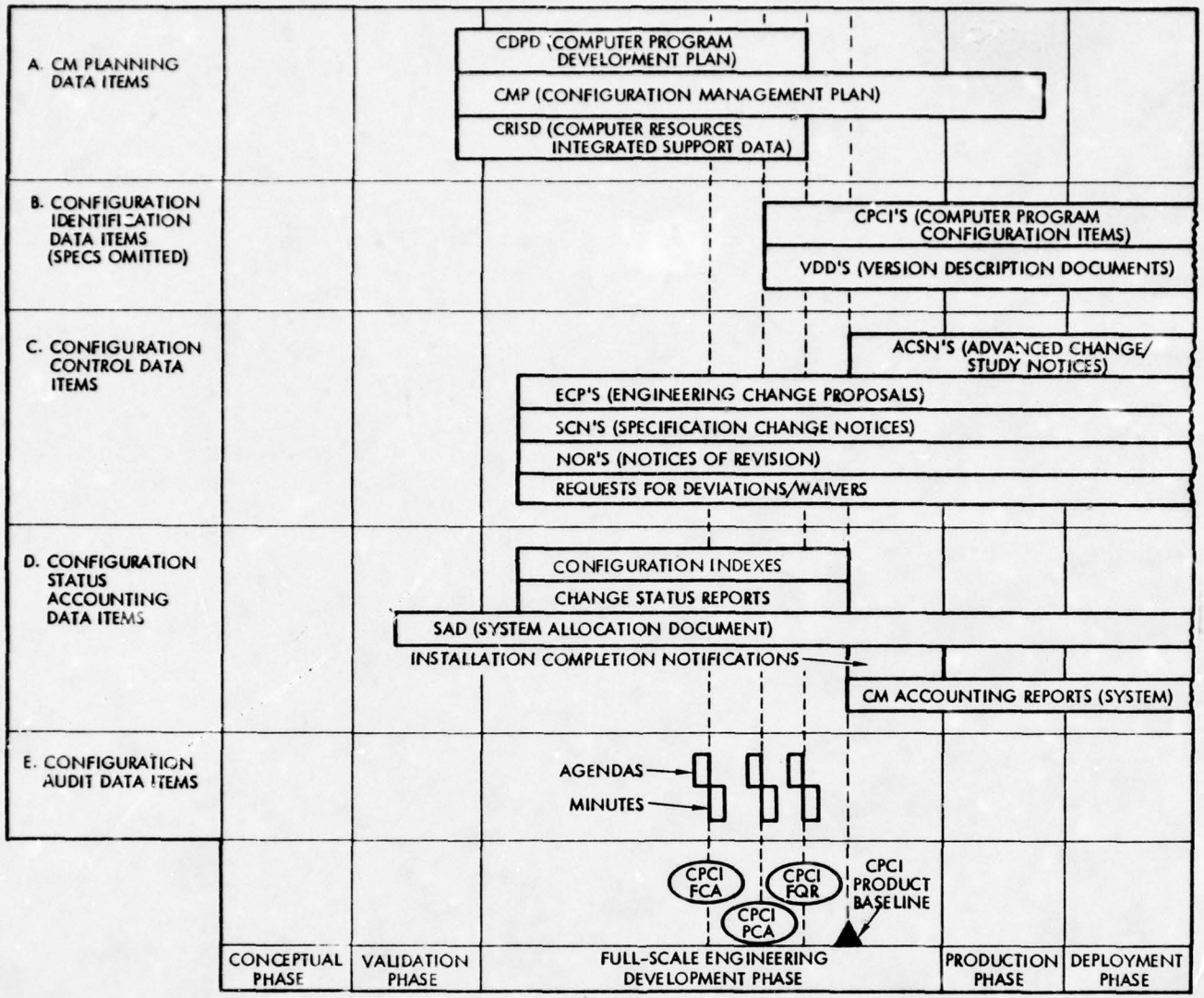



Figure 3-5. Periods of Usage of CM Data Items (Specifications Omitted)

Data Item Name: Computer Program Development Plan (CPDP)

DID No.: DI-S-30567, DI-S-30567A

a. Purpose. Identifies the development contractor's overall plan for management and development of computer programs and related support products. To avoid duplication, the CPDP references the CM Plan and other management plans. It also covers any special aspects of CM not addressed in the CM Plan. Some areas of AFSC prefer to use the basic version (DI-S-30567) of this DID instead of the more detailed version (DI-S-30567A).

b. Origination and Maintenance. Most useful if prepared before the start of the full-scale development phase. Can be prepared by the development contractor as part of his proposal, in response to a requirement in the "Instructions for Preparation of Proposal (IFPP)". Also can be product of validation phase. Should be updated by the development contractor at scheduled intervals in response to a task in the SOW and an entry in the CDRL.

c. Usage. First used by the procuring activity to evaluate the contractor's management approach and methods for computer program development. After contract award, assists the procuring activity in monitoring and evaluating the contractor's actual management of the development and test efforts.

d. Directive and Guidance Documentation.

(1) AFR 800-14, Volume I, and AFSC Supplement 1: paragraph 3m(10).

(2) AFR 800-14, Volume II: paragraph 3-9.

e. Contractor Compliance Documentation.

None.

f. Comments. The CPDP is mandatory for all USAF acquisitions. A CPDP therefore must be prepared by an implementing command that is going to develop its own computer programs. AFSC Supplement 1 (paragraph 3j) to AFR 800-14, Volume I, states that the CPDP should be used to define and monitor the requirement that a contractor make maximum use of existing Government and contractor CM systems.

## CMP

Data Item Name: Configuration Management Plan (CMP)

DID No.: DI-E-3108

a. Purpose. Describes the organizational responsibilities and the procedures to be used by the contractor for implementing the CM requirements stated in the contract.

b. Origination. Prepared by the contractor. Normally a contractor for the full-scale engineering development phase prepares an abbreviated version of the CM Plan as part of his proposal in response to a requirement in the "Instructions for Preparation of Proposal (IFPP)", and then a complete version after contract award, in response to a task in the SOW and an entry in the CDRL. When a single contractor is responsible for both validation and development phases, the contractor should prepare an abbreviated CM Plan as part of the validation phase final report and a complete CM Plan early in the development phase.

c. Usage. Used by the program office first to evaluate the contractor's CM approach and methods and then during the development phase to monitor and evaluate the contractor's CM performance.

d. Directive and Guidance Documentation.

(1) AFSCP 800-7: paragraphs 1-9 and 6-6a; Attachment 3 (paragraph 4); Attachment 5 (paragraph XXXXa).

e. Contractor Compliance Documentation.

(1) MIL-STD-483 (USAF), Appendix I.

f. Comments.

(1) AFSCP 800-7, Attachment 3 (paragraph 4), contains a list of topics suitable for an abbreviated CM Plan to be submitted as part of a proposal. The topics are based on the CM Plan outlined in MIL-STD-483, Appendix I.

(2) Appendix B of this guidebook contains a CM Plan outline that augments the one in Appendix I of MIL-STD-483.

(3) In addition to the contractor CM Plan, an in-house CM Plan by the procuring activity CMO can be advantageous, especially on large or complex development programs.

Data Item Name: Computer Resources Integrated Support Data (CRISD)

DID No.: DI-S-30569

a. Purpose. Identifies the computer resources required to support computer programs being developed by a contractor. Topics covered include support programs and equipment, documentation, configuration management during operation and support, system integration, support personnel, training, maintenance and support verification and validation, support facilities, and the predicted level of computer program changes.

b. Origination and Maintenance. Prepared by the development contractor early in the full-scale engineering development phase in response to a task in the SOW and an entry in the CDRL. A preliminary version of the CRISD is delivered with the development proposal if contractor is required to do so in "Instructions for Preparation of Proposal (IFPP)". The contract may require one or more updates of the CRISD before the development contract is concluded.

c. Usage. Used by the acquisition program's Computer Resources Working Group (CRWG) as input to the Computer Resources Integrated Support Plan (CRISP).

d. Directive and Guidance Documentation. None directly concerning the CRISD. But the following documents discuss the CRISP:

(1) AFSC Supplement 1 (paragraph 3m(5)) to AFR 800-14, Volume I.

(2) AFR 800-14, Volume II: paragraphs 2-4b, 3-8, 3-10, and 10-3.

e. Contractor Compliance Documentation.

None.

## COMPUTER PROGRAMS, DATA, AND PRINTOUTS

Data Item Name: Computer Programs, Data, and Printouts

DID Nos.: DI-E-30145, DI-A-30008

a. Purpose. The items covered by these two DIDs are computer programs and data in machine-readable form on storage media (e.g., punched cards, tape, or disc) and in human-readable form on listings and other printouts. DI-E-30145 is used when the procuring activity wishes to specify formats and other delivery requirements in the CDRL, and DI-A-30008 is used when the details of these requirements are to be defined by contractual program direction.

b. Origination. Prepared by the development contractor. If the DID content requires modification, a backup sheet describing the modification is attached to it.

c. Usage. This data item is the deliverable form of the software product.

d. Directive and Guidance Documentation.

None

e. Contractor Compliance Documentation.

(1) MIL-STD-483, Appendix VI: (paragraph 60.5.5).

f. Comments.

(1) Current AFSC procedure requires that deliverable computer programs and data bases be specified both as data items on the CDRL (DD Form 1423 or AFSC Form 707, 708, or 709) and as line items in the contract schedule.

(2) DID DI-E-30140, Computer Program Product Specification, contains a Section 5, Computer Program Package, that may be helpful in tailoring either of these DIDs.

## VDD

Data Item Name: Version Description Document (Computer Programs)

DID No.: DI-E-3121

a. Purpose: Identifies the exact information on a CPCI tape or deck and all related documents and CPCIs necessary to support the use and regeneration of the CPCI, and records additional data relating to status and use of the CPCI or change.

# Table 3-1. Description of CM Data Items (Specifications Omitted) (Sheet 1 of 2)

b. Origination. Prepared by the releasing contractor or agency to accompany the release of each version, reassembly, or recompilation of a CPCI and the release of each interim Class I change (i.e., a Class I change that occurs between versions, reassemblies, or recompilations).

c. Usage. Provides field personnel and configuration management with a description of the contents of tapes and decks. After PCA, the documentation for each CPCI consists of the Part II specification and the related VDDs. For retrofit CPCI changes, VDDs may be used to supplement information in TCTOs.

d. Directive and Guidance Documentation.

(1) AFR 800-14, Volume II: paragraphs 6-6d and e.

(2) AFSCP 800-7: paragraph 6-6h.

e. Contractor Compliance Documentation.

(1) MIL-STD-483 (USAF), Appendix VIII.

## ACSN

Data Item Name: Advance Change/Study Notice (ACSN)

DID No.: DI-E-3127

a. Purpose. Provides advance information about a contractor's proposed routine changes in contract requirements concerning technical specifications, studies, tests, or other matters. Precedes the detailed effort required for preliminary or formal ECPs (Engineering Change Proposals), for TCPs (Task Change Proposals), or for CCPs (Contract Change Proposals). Not used normally for changes to functional or allocated baselines until after product baseline is established, and not used when emergency, urgent, compatibility, or record type ECPs are contemplated.

b. Origination. Usually prepared by the development contractor, if ACSNs are required by the contract SOW and CDRL. Also may be initiated by the procuring activity.

c. Usage. Used by the procuring activity to screen suggested changes before formal and more costly ECPs, TCPs, or CCPs are prepared. Approval authorizes the contractor to prepare and submit an ECP, TCP, or CCP. ASCNs also can be used to coordinate and integrate related change proposals from associate contractors.

d. Directive and Guidance Documentation.

(1) AFSCP 800-7: paragraph 3-3a (2).

e. Contractor Compliance Documentation.

(1) MIL-STD-483 (USAF), Appendix XIV.

f. Comments. ACSNs may be prepared on AFSC Form 223 or in the contractor's format as approved by the procuring actvity.

## ECP

Data Item Name: Engineering Change Proposal (ECP)

DID No.: DI-E-3128

a. Purpose. To propose changes to a functional, allocated, or product baseline. All such changes are classified either Class I or Class II.

b. Origination. Prepared by the contractor, either on his own initiative or at the direction of the procuring activity. An ECP for a CPCI Class I change normally is submitted in two steps:

(1) A formal ECP first defines the proposed change and its impact, including schedule and cost impact, in sufficient detail to permit authorization for the change. It includes one or more assigned SCN numbers but no actual SCNs.

(2) A revised formal ECP, accompanied by one or more proposed SCNs, defines the exact change. It is submitted following completion of changes to the affected documents and, if the product configuration identification is affected, completion of the design/development/test cycle of the CPCI change. If significant modifications are required in a change following ECP approval, additional revised formal ECPs are submitted.

A preliminary ECP may precede the first formal Class I ECP when all information required for the formal ECP is not available. Furthermore, a contract may require that all Class I ECPs (preliminary or formal) submitted after the product baseline is established be preceded by an Advance Change/Study Notice (ACSN). When proposed changes affect documents for off-the-shelf CPCIs, the required changes to such documents are recorded in a Notice of Revision (NOR) accompanying the ECP. An ECP for a Class II change does not require procuring activity approval prior to implementation unless the contract so specifies. Class II ECPs normally are submitted at the time as, or before, release of the change within the contractor's own facility. Class II ECPs are accompanied by SCNs for any changes to contractor-prepared specifications affected by the ECP changes.

c. Usage. Class I ECPs are used by the procuring activity to evaluate a proposed change, including performance/time/cost effects and impact on the entire system. The first formal ECP provides the basis for authorizing the detailed work (design, development, and test, as required) for the change, and the revised formal ECP provides the basis for approving the completed change and authorizing its incorporation into the CPCI configuration. Class I ECPs also are reviewed by other contractors or agencies whose development activities might be affected. Any NORs (Notices of Revision) accompanying an ECP are used by the procuring activity as a record of changes to documents for off-the-shelf CPCIs. Class II ECPs are used by the procuring activity to verify that the changes are in Class II and not Class I.

d. Directive and Guidance Documentation.

(1) AFR 65-3: Chapter 5.

(2) AFR 800-14, Volume II: paragraph 6-6.

(3) AFSCP 800-7: paragraphs 3-3a, 3-8g, and 6-6n.

e. Contractor Compliance Documentation.

(1) DOD-STD-480A or MIL-STD-481A, and the supplemental requirements of MIL-STD-483 (USAF), Appendix XIV.

f. Comments.

(1) CPCI Class I ECPs governed by DOD-STD-480A are prepared on DD Forms 1692, -1, and -2. DD Form 1692-4 also is used during the production and operational phases to summarize the cost impact of a group of related Class I ECPs containing changes to both computer programs and equipment items. DD Forms 1692-3 and -5 are never used for CPCI changes.

(2) When Class II ECPs governed by DOD-STD-48A are submitted to the procuring activity for concurrence in classification only, which is the usual procedure, the contractor may use either DD Form 1962 (page 1 of the ECP form) or his own form. When the contract requires Class II changes to be approved by the procuring activity prior to implementation, the contractor must use DD Form 1962.

(3) All CPCI ECPs governed by MIL-STD-481A are prepared on DD Form 1693.

(4) DID DI-E-3128 is not an ideal DID for software purposes. For one thing, the term "engineering changes" in its title has wrong connotations for software usage. DI-E-3128 is adequate if it is tailored as follows: (a) add Class II changes to block 3 if ECPs are going to be used for Class II changes, (b) modify the reference to Notices of Revision (NORs) in item 2 of block 10 to make it comply with MIL-STD-483, paragraph 140.14 (see comment 2 under the NOR item in this table for suggested wording), and (c) modify item 3 on milestone charts to make it comply with MIL-STD-483, paragraph 140.13.

## SCN

Data Item Name: Specification Change Notice (SCN) (Computer Program)

DID No.: DI-E-3134

a. Purpose. To propose, transmit, and record changes to a specification.

b. Origination. Prepared by the development contractor. For Class I changes, a separate proposed SCN for each affected specification is submitted to the procuring activity as an enclosure to the revised formal ECP covering the changes. After the proposed SCN is approved, the contractor transmits the approved version and the related change pages of the specification to the specification users. For Class II changes, an SCN for each affected specification is submitted to the procuring activity as an enclosure to the ECP covering the changes at the same time or before the SCN is released within the contractor's own facility. When changes are extensive enough to warrant the issue of a complete revision of a specification, an SCN is not used to transmit it.

c. Usage. Proposed SCNs are used by the procuring activity to evaluate the exact Class I specification changes required by a revised formal ECP. SCNs for Class II changes are used by the procuring activity to verify correct classification. Approved SCNs are inserted by document holders into the affected specifications behind the title page.

d. Directive and Guidance Documentation.

None.

e. Contractor Compliance Documentation.

(1) MIL-STD-490: paragraph 3.3.

(2) MIL-STD-483 (USAF), Appendix VIII.

f. Comments. SCNs are prepared on DD Form 1696. SCNs may be used for changes to other types of documents besides specifications if the procuring activity approves this practice.

## NOR

Data Item Name: Notice of Revision (NOR)

DID No.: None

a. Purpose. Provides an ECP originator with a means of transmitting changes required in another contractor's document because of the ECP. A NOR may be used for CPCI document changes only if the CPCI is an off-the-shelf item and not if it is being developed specifically for a system or has been so developed.

b. Origination. Prepared by the contractor originating the ECP and submitted to the procuring activity with the ECP.

c. Usage. If the ECP is approved, the NOR is used by the procuring activity as a record of changes to the off-the-shelf CPCI's document for immediate and later reference.

d. Directive and Guidance Documentation.

(1) AFR 800-14, Volume II: paragraph 6-6.

e. Contractor Compliance Documentation.

(1) DOD-STD-480A (Section 5) and the supplemental requirements of *MIL-STD-483 (USAF), Appendix XIV (paragraph 140.14).*

f. Comments.

(1) NORs are prepared on DD Form 1695.

(2) Although no Air Force DID exists for NORs, the DID for ECPs (DI-E-3128) includes a requirement for use of NORs. This requirement, however, is based on MIL-STD-480, not MIL-STD-483. The last sentence of paragraph 2 of DI-E-3128 should be modified to read as follows: "When documentation for an off-the-shelf CPCI is affected by the proposed change, a notice of revision (NOR) in accordance with DOD-STD-480A, as modified by MIL-STD-483, shall be included in the ECP *data package.*" *DOD-STD-480A is referenced as the* basic standard because it has superseded MIL-STD-480. Other references to MIL-STD-480 in the DID also should be corrected.

## REQUEST FOR DEVIATIONS/WAIVERS

Data Item Name: Request for Deviations/Waivers

DID No.: DI-E-3129

a. Purpose. Requests and documents a temporary departure from baseline requirements, either before building a configuration item (deviation) or after building it (waiver).

b. Origination. Prepared by the contractor. He requests a deviation prior to coding a CPCI or manufacturing a hardware CI if he wishes wishes to depart from a specified performance or design requirement for a certain version, a certain number of units, or a certain period of time. A waiver is requested when the contractor wishes to deliver a completed configuration item that departs from specified requirements but is considered suitable for use, possibly after rework. At times, the procuring activity may wish to convert a proposed design change (ECP) to a deviation or vice versa.

c. Usage. Used by the procuring activity to evaluate the proposed deviation or waiver.

d. Directive and Guidance Documentation.

(1) DODI 5010.21: paragraph V.F.

(2) AFSCP 800-7: paragraph 3-3b.

e. Contractor Compliance Documentation.

(1) DOD-STD-480A or MIL-STD-481A.

f. Comments. Deviations and waivers governed by DOD-STD-480A may be prepared on DD Form 1694 or on the contractor's own form, if approved. When governed by MIL-STD-481A, they must in most cases be prepared on DD Form 1694.

## CONFIGURATION INDEX

Data Item Name: Configuration Index (Computer Program)

DID No.: DI-E-3122

a. Purpose. Presents the current status of CPCI development in terms of specifications and other documents that depend on the CPCI configuration, such as test plans and procedures, user manuals, and the Version Description Document. Reports (a) the basic issue and each complete revision of each document, (b) all ECPs and SCNs subsequently incorporated into the documents, (c) impact of ECP/SCN changes on related CPCIs, CIs, or documents, (d) approved *ECPs not yet incorporated,* and (e) a summary record of CPCI development, test, audit, and qualification milestones. A group of interrelated CPCIs may be combined in a single index.

b. Origination. Prepared and issued by the development contractor throughout the full-scale engineering development phase. Initial issue with title page, Section A (Configuration Item Development Record), and Section I (Part I Development Specification) is due within 30 days after establishment of the allocated baseline (or functional baseline, if the allocated baseline is not employed). Subsequent issues are published monthly or at other regular intervals established by the procuring activity, with sections added as needed to list additional documents published. Always is accompanied by the Change Status Report, which gives the current status of all ECPs to the CPCI.

c. Usage. Used by the procuring activity to monitor development and maintenance of CPCI technical documentation and to provide general visibility of CM activities and events during development.

d. Directive and Guidance Documentation.

(1) AFSCP 800-7: paragraphs 2-11e, 2-12e, and 2-14e.

e. Contractor Compliance Documentation.

(1) MIL-STD-483 (USAF), Appendix VIII.

## CHANGE STATUS REPORT

Data Item Name: Change Status Report (Computer Program)

DID No.: DI-E-3123

a. Purpose. Supplements the Configuration Index by providing detailed status of all proposed changes (ECPs) to the CPCI documents listed in the Configuration Index.

b. Origination. Prepared by the development contractor throughout the full-scale engineering development phase. First issued after contractor assigns a number to the first ECP to be prepared against a document listed in the Configuration Index. Thereafter it is always published concurrently with the Configuration Index.

c. Usage. Used by the procuring activity and the contractor to determine current status of CPCI document changes.

d. Directive and Guidance Documentation.

None.

e. Contractor Compliance Documentation.

(1) MIL-STD-483 (USAF), Appendix VIII, paragraph 80.11. (Note: The heading and text of paragraph 80.11 erroneously refer to the Change Status Report as the "Change Status Listing." The Status Listing is Section I of the Change Status Report. It is described in paragraph 80.11.1.1 and illustrated in Figure 14.)

## SAD

Data Item Name: System Allocation Document (SAD)

DID No.: DI-E-3116

a. Purpose. Identifies the total collection of configuration items that make up a multi-location system, and identifies the location of all CIs by CI serial number. Originally used for systems in fixed installations, but also can be used for flight systems.

tinually updated. A SAD is maintained until the end of system testing.

c. Usage. Used by the procuring activity and contractors to support delivery, integration, and system testing at each location.

d. Directive and Guidance Documentation.

(1) AFSCP 800-7: paragraphs 1-16 and 6-6k.

e. Contractor Compliance Documentation.

(1) MIL-STD-483 (USAF), Appendix XI.

f. Comments. The SAD is an optional document whose function may be performed by status accounting reports or other kinds of reports.

## ICN

Data Item Name: Installation Completion Notification

DID No.: DI-E-3107

a. Purpose. Reports accomplishment of Class I updating/retrofit changes (via ECPs TCTOs) by a contractor's test and field organizations after establishment of the product configuration baseline and before the end of Development Test and Evaluation (DT&E).

b. Origination. The notification form is originated by the contractor responsible for preparing retrofit kits or modifications and is sent with the retrofit kit or modification instructions to the procuring activity or contractor activity having custody of the CI affected. After accomplishing the required change, the procuring activity or contractor activity having custody of the CI completes the notification form and distributes copies.

c. Usage. One copy of the completed notification form is sent to the contractor responsible for preparing configuration status accounting reports for the procuring activity. It is used to update the TCTO/ECP status of delivered CIs. Other copies may be sent to the procuring activity CMO and associate contractor CMO, if required. The activity accomplishing the change files the original with the CI records.

d. Directive and Guidance Documentation.

None.

e. Contractor Compliance Documentation.

(1) MIL-STD-483 (USAF), paragraph 3.11 and Appendix XV.

## STATUS ACCOUNTING REPORTS (CII AND CSAR)

Data Item Name: CM Accounting Reports (Machine or Manually Prepared)

DID No.: DI-E-3133

a. Purpose. These reports usually consist of a Configuration Identification Index (CII) that documents the approved configuration of a system and a Configuration Status Accounting Report (CSAR) that documents the current configuration of the system.

b. Origination. Prepared by the contractor responsible for status accounting for an entire system, usually after the system product baseline is established. Other contractors and agencies participating in the program are required to submit original configuration data and update data to the status accounting contractor for inclusion in these reports.

c. Usage. These reports are the primary means by which management and implementing personnel keep track of system baseline configurations and changes to those configurations and coordinate the many tasks required by the changes.

d. Directive and Guidance Documentation.

(1) AFR 65-3: Chapter 4 and Appendix F.

(2) AFR 800-14, Volume II: paragraph 6-7.

(3) AFSCP 800-7: Chapter 4 and other parts.

e. Contractor Compliance Documentation.

(1) MIL-STD-482A.

Data Item Name: Agenda – Design Reviews, Configuration Audits and Demonstrations

DID No.: DI-A-3029

a. Purpose. Notifies the procuring activity of forthcoming design reviews, CM audits, and demonstrations and states the purpose and objectives to be accomplished. The CM audits are the FCA (Functional Configuration Audit) and the PCA (Physical Configuration Audit). The FQR (Formal Qualification Review) also is a CM audit to the extent that its accomplishments are documented in the Configuration Item Development Record (part of the Configuration Index).

b. Origination. Prepared and distributed by the development contractor in accordance with the master milestone schedule, and coordinated with the procuring activity.

c. Usage. Used by the procuring activity to plan for a review or audit and to review any data that will be a subject of the review or audit.

d. Directive and Guidance Documentation.

(1) AFR 65-3: Chapter 5.

(2) AFR 800-14, Volume II: paragraphs 4-9 and 6-8.

(3) AFSCP 800-7: Chapter 5 and paragraph 6-6, item 1.

e. Contractor Compliance Documentation.

(1) MIL-STD-483, paragraph 3.9 and Appendix XII.

(2) MIL-STD-1521A.

## MINUTES

Data Item Name: Minutes of Formal Reviews, Inspections and Audits

DID No.: DI-E-3118

a. Purpose. Provide a record of reviews, inspections, and audits of a system or configuration item. The CM audits are the FCA and PCA. The FQR also is a CM audit to the extent that its accomplishments are documented in the Configuration Item Development Record (part of the Configuration Index).

b. Origination. Prepared by the contractor and distributed in accordance with the CDRL.

c. Usage. After receiving the minutes, the procuring activity provides a formal, written acknowledgement of the review/audit accomplishments to the contractor. Both contractor and procuring activity perform followup on all action items listed in the minutes.

d. Directive and Guidance Documentation.

(1) AFR 65-3: Chapter 5.

(2) AFR 800-14, Volume II: paragraphs 4-9 and 6-8.

(3) AFSCP 800-7: Chapter 5 and paragraph 6-6, item 1.

e. Contractor Compliance Documentation.

(1) MIL-STD-483: paragraph 3.9 and Appendix XII.

(2) MIL-STD-1521A.

6) Assisting in classifying deviations and waivers.

7) Evaluating documentation maintenance systems, including incorporation of changes.

Program offices can delegate direct contract-related CM functions to the CAS activity responsible for in-plant monitoring of the contractor's Government work. Contract Administrative Services are provided by the Air Force Contract Management Division (AFCMD), the Defense Contract Administrative Service (DCAS), and other DOD agencies. AFPROs (AF Plant Representative Offices) are the field extensions of the AFCMD.

c) Monitoring CM Through Other Program Office Areas. Some information on the conduct of a contractor's CM program may be available from the program office QA area. If the contractor is required by MIL-S-52779 (AD) to perform QA audits of his CM program and to take other QA measures to ensure the CM program is working properly, documentation of these QA activities may give the program office CM manager useful insights. Other program office areas, particularly system engineering and V and V (verification and validation), also are concerned with the functions and products of CM and may observe problems not apparent to the CM manager.

d) Monitoring CM through Configuration Audits. The Functional Configuration Audit (FCA), Product Configuration Audit (PCA), and sometimes the Formal Qualification Review (FQR) are the CMO's ultimate evaluation of a contractor's product. These events are described in subsection 3.1.4.

e) CM Evaluation Checklists. Checklists are helpful devices for evaluating matters that are subject to many criteria or difficult criteria. They are particularly useful for repetitive evaluation tasks, such as evaluation of ECPs. Government RSS on CM contain a number of useful checklists, as well as other lists and descriptions that can be turned into checklists easily. Some of these checklists and potential checklists are:

1) FCA and PCA preparation checklists in MIL-STD-1521A.

2) ECP preparation checklists in DOD-STD-480A, as modified by MIL-STD-483.

3) ECP evaluation checklists (see 5.2.4 of this guidebook for references).

4) ECP checklist for verification of Class I classification in AFSCP 800-7 (Figure 3-1). Several of these criteria are hardware-oriented and not applicable to software.

5) CM Field Inspection Checklist (AFSC Form 408) is discussed in AFSCP 800-7, paragraph 1-17c.

6) Computer Resource Manager's Checklist based on AFR 800-14, Volumes I and II, are available from HQ AFSC/XRF.

7) Attachments 3, 4, and 5 of AFSCP 800-7 on CM requirements in RFPs and contracts.

# 4. SPECIFIC GUIDANCE FOR CONFIGURATION IDENTIFICATION

Configuration identification in this section covers not only the subject of configuration identification documentation, but also other controlled documents, CPCI definition, specification trees, and item identifiers.

## 4.1 COMPUTER PROGRAM CONFIGURATION ITEMS (CPCIs)

Identifying the computer program configuration items for a system or system segment procurement is a system engineering task rather than a CM task but it has such an important bearing on CM that it is discussed in this section. It is "configuration identification" in a broader sense than is usually thought of for that term and must occur before the major part of the configuration identification documentation activity can begin.

This subsection discusses the place and importance of the CPCI in the system hierarchy and presents some principles for CPCI identification.

### 4.1.1 Defining the System Hierarchy and Breakdown

To understand and keep track of the organizational relationships of CPCIs and hardware CIs within a computer resource configuration and the overall system, it is necessary to define the hierarchy of levels in the system. The levels most commonly mentioned in Government documents for this purpose are the following:

a) System. "A composite of items, assemblies (or sets), skills and techniques capable of performing and/or supporting an operational (or nonoperational) role. A complete system includes related facilities, items, material, services, and personnel required for its operation to the degree that it can be considered a self-sufficient item in its intended operational (or nonoperational) and/or support environment." (AFSCP 800-7)

b) System Segment. "A discrete package of system performance requirements, functional interfaces, and CIs contracted to one contractor or assigned to one Government organization directly responsible to the procuring activity for that part of a system's total performance." (MIL-STD-483) Segment is a special level that is used "when a system or major equipment is acquired on an incremental or evolutionary basis or when a segment(s) of an existing system

is to undergo major modification." (MIL-STD-483) It also may be required "when the procurement of a large system is apportioned among different program offices" or "when the computer resources or computer program portions of the system are separately developed or contracted." (AFSCP 800-7). It sometimes is equivalent to a "subsystem" or "functional area" but at other times it contains within it more than one subsystem or functional area. Consists of hardware CIs, CPCIs, or both.

c) Computer Program Configuration Item (CPCI). An aggregation of computer programs that satisfies an end-use function and is designated by the Government for configuration management. (Paraphrased from AFSCP 800-7). Usually referred to in MIL-STD-490 and MIL-STD-483 as "computer programs." Corresponds to hardware configuration items (CIs).

d) Computer Program Component (CPC). "A functionally or logically distinct part of a computer program configuration item (CPCI) distinguished for purposes of convenience in designing and specifying a complex CPCI as an assembly of subordinate elements." (MIL-STD-483) Sometimes referred to in MIL-STD-490 and MIL-STD-483 as "subprograms."

e) Routine. Lowest compilable or assemblable element of a computer program. Also commonly called a "subroutine".

The system level is defined by HQ USAF, the segment and CPCI levels usually by the procuring activity, and the CPC and routine levels by the development contractor.

Complex systems usually require additional levels beyond those described above, both above the CPCI level and below it. For example, "subsegments" might be necessary to permit grouping of CPCIs within a segment. "Functional areas" or "subsystems" also are used as terms for groups of CPCIs. Software development contractors frequently use one or more additional levels, usually between CPCs and routines, and call them "modules," "submodules," "functions," "elements," or similar terms.

After system requirements have been allocated down to the routine level, a complete chart of the system breakdown can be prepared. A portion of such a chart is shown in Figure 4-1, with the hierarchy levels stated at the right.

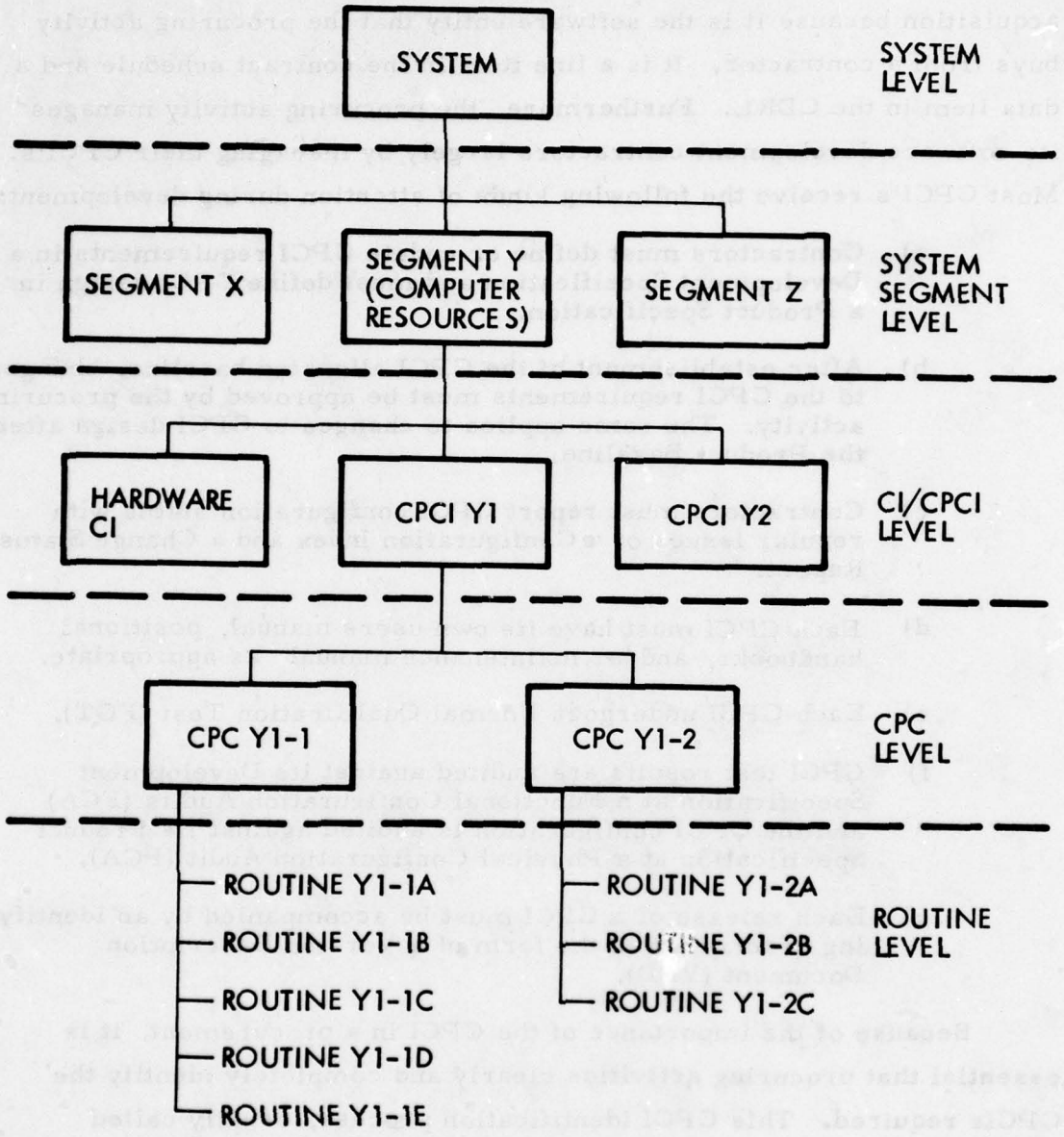



Figure 4-1. Portion of a System Breakdown

### 4.1.2 The Importance of CPCIs

"CPCI" is an important designation in computer resources acquisition because it is the software entity that the procuring activity buys from a contractor. It is a line item in the contract schedule and a data item in the CDRL. Furthermore, the procuring activity manages its software development contractors largely by managing their CPCI's. Most CPCI's receive the following kinds of attention during development:

a) Contractors must define or update CPCI requirements in a Development Specification and must define CPCI design in a Product Specification.

b) After establishment of the CPCI allocated baseline, changes to the CPCI requirements must be approved by the procuring activity. The same applies to changes to CPCI design after the Product Baseline.

c) Contractors must report CPCI configuration status with regular issues of a Configuration Index and a Change Status Report.

d) Each CPCI must have its own users manual, positional handbooks, and/or maintenance manual, as appropriate.

e) Each CPCI undergoes Formal Qualification Test (FQT).

f) CPCI test results are audited against its Development Specification at a Functional Configuration Audits (FCA) and the CPCI configuration is audited against its Product Specification at a Physical Configuration Audit (PCA).

g) Each release of a CPCI must be accompanied by an identifying description in the form of a Version Description Document (VDD).

Because of the importance of the CPCI in a procurement, it is essential that procuring activities clearly and completely identify the CPCIs required. This CPCI identification process, usually called "CPCI selection" in Government documents, has two basic steps: (a) identification of the total set of deliverable software processes or functions and (b) grouping of these processes or functions into CPCIs.

### 4.1.3 Identifying Deliverable Software Items

To identify contractually deliverable software, the following three rules generally apply:

a) Include in the contract as deliverable items all operational software and all test and support software, including firmware, data, and proprietary items, that are required to cost effectively use, operate, modify, or maintain the system over its life cycle.*

b) When the cost effectiveness of a required item of test or support software cannot be determined, include in the contract an option to acquire it later.**

c) Test or support software that is required only during development need not be specified as contract deliverables unless its development or use has strong direct effect on items designated as deliverables.

The three categories of programs mentioned in these rules (operational, test, and support) are the categories recognized by AFR 800-14, Volume II (paragraph 10-5), and described there as follows:

a) Operational Computer Programs. Required to operate the system. Are loaded and run in the computer equipment during system operation. Include executive/supervisor programs; functional/application programs; and input/output programs.

b) Test Computer Programs. Used to analyze or test system and component performance. May be integrated into the operational computer programs as test CPCs that operate concurrently with system operation. Include maintenance/diagnostic programs.

c) Support Computer Programs. Used generally for the development and maintenance of other computer programs. Include operating systems, assemblers, compilers, and loaders. In training devices, for example, these programs include preflight check programs, data base modification programs, and student performance data printout programs.

---

*This rule is based on the following Government documents:

DODD 5000.29: paragraph V.E
AFR 800-14, Volume I: paragraph 3i
AFSC Supplement 1 to AFR 800-14, Volume I: paragraphs 3i and 3m(8)

**This rule is based on the following Government document:

AFSC Supplement 1 to AFR 800-14, Volume I: paragraph 3i

Operating system elements required during system operation belong in the first group, and operating system elements required only during development and maintenance belong in the third one.

Computer data is not mentioned in the AFR 800-14 paragraph cited above. However, data bases, test input data, and other kinds of non-executable data required in the use or maintenance of a system sometimes are managed more effectively if identified separately from the executable portions of a software system.

### 4.1.4 Grouping Deliverable Software into CPCIs

The second step of the CPCI identification process is to group the deliverable software processes into CPCIs*. This is not a simple matter on a large system because of the many possible arrangements and the absence of any firm criteria. Some of the desirable functional and administrative characteristics of a CPCI can be defined, however:

a) All of its processes are of the same type, either operational, test, or support.

b) It groups together processes that interact extensively or in complex ways.

c) All of its processes are intended to run on the same computer or computers.

d) It can be developed and tested feasibly by a single contractor.

e) Its processes are all needed at about the same time in the acquisition schedule.

f) All of its processes have a similar criticality to system operation.

g) All of its processes have a similar development risk.

---

*This step is required by the following Government documents:

DODD 5000.29: paragraph V.C
DODD 5010.19: paragraph III.C
DODI 5010.21, Enclosure 1: paragraph 6
AFSC Supplement 1 to AFR 65-3, Appendix F: paragraph 1a
AFR 800-14, Volume I, Attachment 1: paragraph 4
AFR 800-14, Volume I: paragraph 3m(8)
AFSC Supplement 1 to AFR 800-14, Volume I: paragraph 3m(8)

h) All of its processes are subject to the same degree of development control (including CM).

i) All of its processes are subject to the same operational and maintenance concepts.

j) All of its processes are required at each installation where it will be used.

k) It is small enough to be monitored during development by a single Program Office representative.

Not all of the above goals* usually are achievable in a CPCI. Frequently the technical and administrative objectives for CPCI selection are in conflict with each other, with the technical inclination being toward a greater number of smaller CPCIs to improve technical visibiiity, while the administrative need is for fewer CPCIs and CPCI interfaces so that interface documentation and control and integration testing do not become unduly complex.

For large software programs, such opposed requirements can be balanced through optimization analysis. First, figures of merit are established for the various technical and administrative factors involved. Then function allocations are varied systematically among different numbers of CPCIs to determine the most advantageous grouping. Optimum grouping may result in a variety of sizes and complexity levels, including some small CPCIs for distinctively different types of programs, such as test or support programs. Data elements may be combined with executable computer programs or may be given separate CPCI identification, depending on their characteristics.

Preliminary selection of CPCIs should be performed by system engineering in the conceptual phase, with the needs and recommendations of all participants considered, including those of supporting and using commands as defined in the PMP and other documents. This preliminary CPCI list should remain open to refinement until system design is completed, however. Development phase contractors may be able to improve it.

---

*This list of CPCI characteristics is adapted and expanded from a list in AFSCP 800-7, paragraph 6-3b.

### 4.1.5 Special Problems: Multiple Locations and Modified CPCIs

When a CPCI requires two or more slightly different configurations because of operational variations at different locations (i.e., variations in executable logic, not merely different data base adaptation data), each configuration may be identified either as a separate CPCI or as a different type of the same CPCI, depending on the degree of difference. The documentation requirements for these two cases are different. If the different configurations are to be considered different types of one CPCI, the types are documented within the same Development and Product Specifications in accordance with paragraphs 4.1.2 and 4.3b of MIL-STD-490. If they are to be considered as separate CPCIs, however, separate Development and Product Specifications usually are prepared for each CPCI. An alternative course for separate CPCIs is to prepare a Development Specification and a Product Specification for one CPCI and Addendum Specifications for the remaining ones.

A similar problem occurs when a CPCI is modified while in operational use. If the modification is extensive, re-identification as a new CPCI and redocumentation in new Development and Product Specifications should be considered as a possible course, rather than merely updating the existing specifications. Any one of the following situations would strongly recommend such a course:

a) New design reviews are needed.

b) One or more additional modules are required in the CPCI.

c) Major reprogramming is needed.

d) New specifications are needed anyhow to incorporate many previous changes.

e) Both versions of the CPCI will be used.

### 4.1.6 Defining Form of Software Deliverables

After the CPCIs for a particular procurement are defined, the form of each delivered CPCI must be established for inclusion in the CDRL. The CDRL entry for each CPCI should specify both the state of the code (source code, object code, or load module) and the medium on which the code will be recorded (punched card deck, magnetic tape,

punched mylar tape, etc). The exact requirements for each CPCI will depend on the Government's planned usage following delivery.

Normally, the complete CPCI source code on magnetic tape or card deck or both, suitable for compilation or assembly, and the complete CPCI object code on magnetic tape or card deck or both, suitable for loading and execution in the operational computers, are required. Sometimes location-specific tapes of cataloged operational programs also are required for direct incorporation into system disks at different locations. For systems with batch capabilities, batch job control decks also are helpful. At least one copy of a complete listing for each delivered software item should be specified in the CDRL. For large files, a tape log, or summary listing, also is helpful.

Sometimes vendors supplying proprietary off-the-shelf software items, such as operating systems, will release only the object versions of their software. In such cases, arrangements are made to have the vendor continue maintenance of his product throughout the system life cycle, since maintenance cannot be accomplished without the source code.

Each CDRL entry for a CPCI should include a reference to one of the following two Data Item Descriptions (DIDs): DI-E-30145, "Computer Software/Computer Program/Computer Data Base Configuration Item(s)," or DI-A-30008, "Computer/Machine Products (Special)." (See Table 3-1 for descriptions of these DIDs.) Backup sheets should be prepared for CPCIs that require some modification of the referenced DIDs. In addition to its CDRL entry, each CPCI must be included in the contract schedule as a line item.

The contractor should specify the physical form, dimensions, and material of the CPCI delivery media in Section 5 (Preparation for Delivery) of the Product Specification.

## 4.2 CONFIGURATION IDENTIFICATION DOCUMENTS

Configuration identification documents embody the requirements and design of software products in increasing detail and accuracy as a development effort proceeds. During most of the period of software

planning and development, these identification documents are the major objects of CM attention, since the software itself is not ready for any control until the last quarter or so of this period.

Program Office CM personnel often are given the responsibility of selecting and tailoring the DIDs to be used for preparation of configuration identification documents. They also should review the completed identification documents, as well as other deliverable documents, to ensure that they agree with MIL-STD-490, MIL-STD-483, and any other applicable standards and that the applicable documents listed in Section 2 of each document are really applicable to program requirements and have the correct issue and date shown.

Eight of the most commonly used types of configuration identification documents for software procurements are the following:

a) System Specification

b) System Segment Specification

c) Computer Program Development Specification

d) Computer Program Product Specification

e) Interface Control Drawing, or Interface Design Specification

f) Data Base Specification

g) Addendum Specification

h) Inventory item Specification

All of these documents usually are subject to baseline configuration control except for the Interface Control Drawing and Interface Design Specification, which are subject to interface control.

The Development Specification and Product Specification are considered by MIL-STD-483 to be parts of a single specification, so they are given the same specification number and are actually labelled "Part I" and "Part II." The justification and consequences of this practice are stated in MIL-STD-490, paragraph 3.1.4: "This practice requires both parts for a complete definition of both performance requirements and detailed design requirements governing fabrication

[i.e., coding]. Under this practice, the development specification remains alive during the life of the item as the complete statement of performance requirements. Proposed design changes must be evaluated against both the product fabrication [i.e., coded design] and the development parts of the specification." In its classification scheme for specifications, MIL-STD-490 uses the designations "B5" and "C5" for these same two computer program specifications.

The purpose, origination, maintenance, usage, and reference documentation for each of the eight documents listed above, as well as additional information on the selection and preparation of specifications, is given in the SAE Guidebook for Computer Program Documentation Requirements.

## 4.3 OTHER CONTROLLED DOCUMENTS

In addition to the configuration identification documents discussed in the preceding subsection, a number of other documents require CM attention during planning and some measure of configuration control (usually only contractor internal control) during certain stages of the development process. The following documents usually are included in this category:

a) Test Plans and Procedures. Provide detailed instructions for testing the software and reporting test results.

b) User Manual. Provides instructions for operational use of a software product.

c) Positional Handbooks. Describe procedures for operating a computer and executing the computer programs.

d) Computer Programming Manual. Provides programming information required for maintenance or modification of the software.

These documents are discussed in the SAE Guidebook for Computer Program Documentation Requirements.

## 4.4 SPECIFICATION TREES

A specification tree is prepared by assigning appropriate types of specifications to the various levels of the system hierarchy (discussed in paragraph 4.1.1) and then adding these specification types to the

system breakdown (a system breakdown is shown in Figure 4-1). The result is illustrated in Figure 4-2.

Such a logical representation of system elements is useful in assessing the impact of a change at any level, as well as in scheduling and for other program management purposes. A tree of this type may be expanded to include other system documentation, and the documentation numbers may be added to the boxes. A specification tree or a complete documentation tree are suitable items for inclusion in paragraph 3.4 ("Documentation") of the System Specification.

## 4.5 ITEM IDENTIFIERS

A configuration management program cannot function unless the items to be controlled are clearly and uniquely identified at all times. This applies to CPCIs, routines, and other controlled software elements, to the tapes, decks, and other media on which the software elements are stored, and to the specifications and other controlled documents associated with the software. The importance of proper item identifiers is not limited to CM, of course, for they are essential to all areas and all stages of development, operation, and support.

In general, the Government is responsible for assigning identifiers to software items at the CPCI level and above, in accordance with current policy, and the development contractor is responsible for assigning identifiers to items below the CPCI level, in accordance with applicable document standards or, when these do not cover an item, the contractor's own standards.

### 4.5.1 Qualities of Item Identifiers

Qualities that item identifiers on software acquisition programs may possess include the following:

a) Uniqueness. The most important quality. Every unique item requires a unique identifier so that identification will be absolute. (Even some identical items require unique serial numbers for allocation and provisioning purposes.) Some identification systems, such as the Air Force CPIN (Computer Program Identification Number) System, offer item identifiers that are unique within the entire Air Force. For routines and other lower level software elements, however, it is usually sufficient that identifiers be unique within the system. Any alphanumeric characters are suitable for establishing uniqueness.

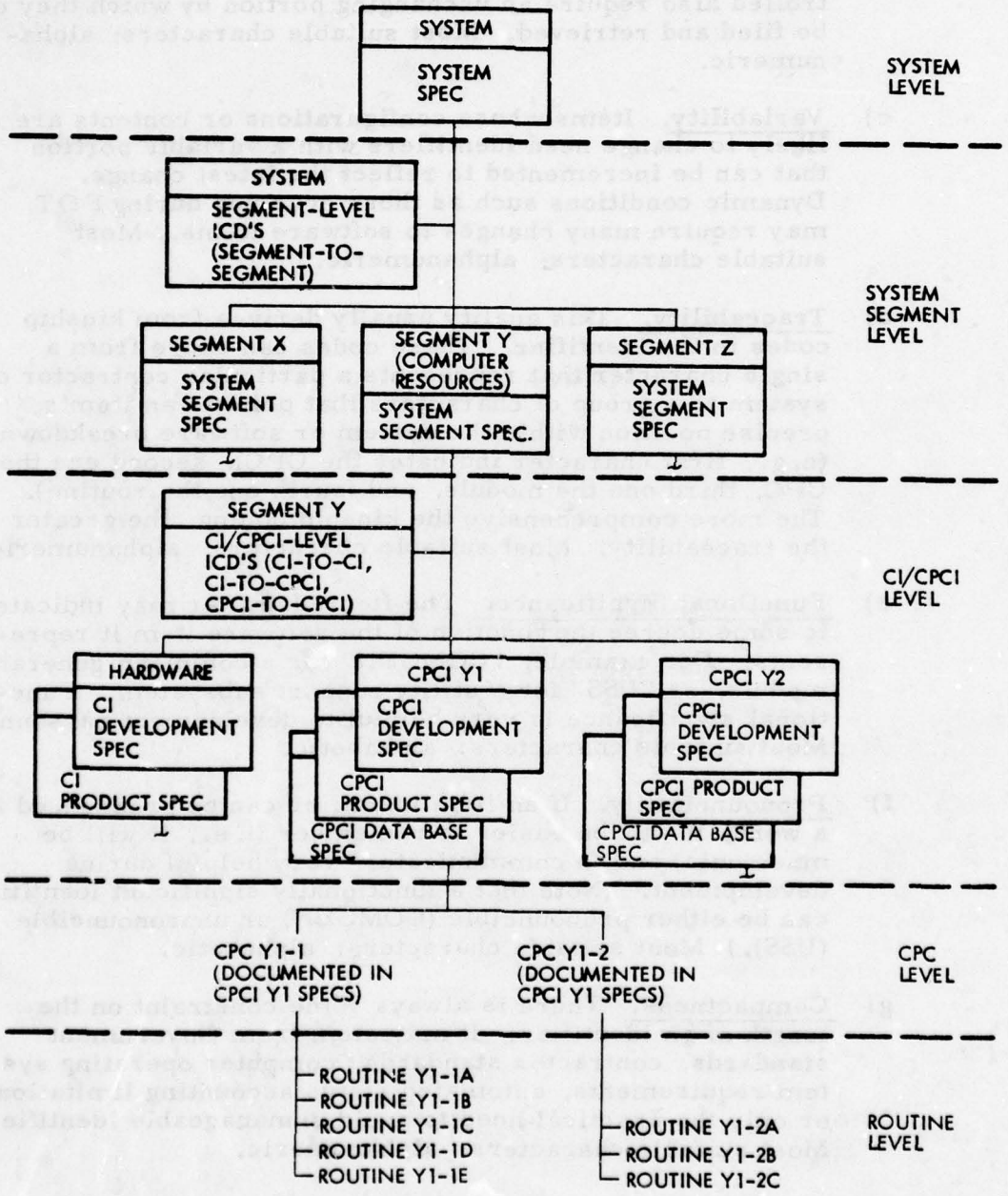



Figure 4-2. Portion of a Specification Tree

b) Retrievability. The identifier for each callable software item must include an unchanging portion that will be the item's name or label within the computer system. Identifiers for documents and all other items to be controlled also require an unchanging portion by which they can be filed and retrieved. Most suitable characters: alphanumeric.

c) Variability. Items whose configurations or contents are likely to change need identifiers with a variable portion that can be incremented to reflect the latest change. Dynamic conditions such as those ocurring during FQT may require many changes to software items. Most suitable characters: alphanumeric.

d) Traceability. This quality usually derives from kinship codes in the identifier. These codes can range from a single character that represents a particular contractor or system to a group of characters that point to an item's precise position within the system or software breakdown (e.g., first character indicates the CPCI, second one the CPC, third one the module, and fourth one the routine). The more comprehensive the kinship coding, the greater the traceability. Most suitable characters: alphanumeric.

e) Functional Significance. The item identifier may indicate to some degree the function of the software item it represents. For example, "COMGEN" for a command generation module, or "USS" for a utility support subsystem. Functional significance is very helpful to development personnel. Most suitable characters: alphabetic.

f) Pronouncibility. If an item identifier can be pronounced as a word, it will be easier to remember (i.e., it will be mnemonic) and to communicate. Very helpful during development. (Note that a functionally significant identifier can be either pronouncible (COMGEN) or unpronouncible (USS).) Most suitable characters: alphabetic.

g) Compactness. There is always some constraint on the length of an identifier. It may stem from Government standards, contractor standards, computer operating system requirements, automated status accounting limitations, or only the practical need to avoid unmanageable identifiers. Most suitable characters: alphanumeric.

The first two qualities – uniqueness and retrievability – are mandatory ones for all item identifiers, and the last one – compactness – usually is an important consideration. Some item identifiers can serve very well with no more than these three qualities.

Traceability of an item from its identifier also is essential but it need not rely on pointers within the identifier itself. Indexes or tables can provide this information for either manual or automated lookup.

Additional qualities should be considered according to the functions to be performed by the identifiers. Functional significance and pronouncibility are desirable qualities when human beings must deal with a multitude of similar items, for example, but are irrelevant and wasteful if an automated system is going to be processing the identifiers.

No item identifier can embody all seven of the above qualities to a high degree, primarily because the last one – compactness – precludes it. Compromises are necessary.

Sometimes several identifiers may be needed for the same item in order to satisfy the different needs of the various systems within which the item exists. A single document thus may be required to bear a Government specification number, a project identification number, a contractor classified document number, a CDRL sequence number, and other identifiers.

### 4.5.2 Government Item Identification Requirements

Software development item identification requirements appearing in Government regulations, specifications, and standards are summarized in Table 4-1. In addition to documentation, software elements, and software storage media, the table includes configuration control and status accounting data items whose identifiers are defined in Government documents.

For computer program documents and CPCIs, two different sets of item identification requirement documents are listed. In the first set, MIL-STD-490 is the key document for document identifier requirements and MIL-STD-483 is the key document for CPCI identifier requirements. In the second set, AFLC Supplement I to AFR 800-14, Volume II, is the

Table 4-1. Summary of Government Requirements for Item Identifiers (Sheet 1 of 3)

| Items Requiring Identifiers | Directive and Guidance Documents | Contractor Compliance Documents | Major Characteristics of Item Identifiers | Party Responsible for Assigning Item Identifiers | Comments |
|---|---|---|---|---|---|
| A. DOCUMENTATION | | | | | |
| Computer Program Documents (Including Specifications) | • AFR 65-3, para. 2-6a<br>• AFSCP 800-7, paras. 2-11d, 2-12d | • MIL-STD-490, para. 3.2.1o<br>• MIL-STD-482, page II-14 | 1. Contractor code identification<br>2. 15 alphanumeric characters maximum plus alphabetic revision letter (excluding I, O, Z, etc.) | Cataloguing Handbook H4-1<br>Contractor | This document identifer can be applied to all other computer program documents, in addition to specifications |
| | • AFR 800-14, Vol. II, para. 6-5b, & AFLC supp. 1, paras. 6-5b(2), 6-5b(3), 7-10c, and Chapter 10.3 | • None | 17 to 19 alphanumeric characters plus version identifier. Is based on CPCI CPIN (see Sheet 2). | AFLC (OC-ALC) | These identifiers are CPINs (Computer Program Identification Numbers). (See CPCI CPINs on Sheet 2.) |
| Flow Charts and Program Listings | • AFSCP 800-7, para. 6-6i(4) | • None | 1. Identify CPCI flow charts with CPCI number.<br>2. Identify CPC flow charts with CPC number (add revision code suffix for detailed flow charts affected by CPC revisions.)<br>3. Identify CPCI listings with CPCI number.<br>4. Identify CPC listings with CPC number, including revision code suffix. | Contractor | Characteristics of CPC flow charts and listings also apply to flow charts and listings of routines and other controlled elements of a CPCI. |
| VDDs (Version Description Documents) | • AFSCP 800-7, para. 6-6h(4) | • MIL-STD-483, paras. 80.12.1.1(e), 80.12.1.1(f) | "VDD" followed by identification of CPCI version and, if applicable, the interim change number. | Contractor | |

## Table 4-1. Summary of Government Requirements for Item Identifiers (Sheet 2 of 3)

| Items Requiring Identifiers | Directive and Guidance Documents | Contractor Compliance Documents | Major Characteristics of Item Identifiers | Party Responsible for Assigning Item Identifiers | Comments |
|---|---|---|---|---|---|
| B. SOFTWARE ELEMENTS | | | | | |
| CPCIs (Computer Program Configuration Items) | • AFSCP 800-7, para. 6-6i(1) | • MIL-STD-483, para. 90.3.2.3<br>• MIL-STD-482, pages II-3, II-4 | 1. Unchanging base with 7 alphanumeric characters.<br>2. Variable suffix with 3 alphanumeric characters.<br>3. CPCI copies at different locations need additional notation. | Contractor | |
| | • AFR 800-14, Vol. II, para. 6-5b, and AFLC supp. 1, paras 6-5b(2), 6-5b(3), 7-10c, and Chapter 16.3. | • None | 1. 17 to 19 alphanumeric characters showing aerospace category (2 characters), subcategory (1) 3 system or equipment designator (up to 9), subsystem (1 or 2), type of program or documentation (2), and basic programs (2 or 3).<br>2. Needs separate version identifier. | AFLC (OC-ALC) | These identifiers are CPINs (Computer Program Identification Numbers). (See documentation CPINs on Sheet 1.) |
| CPCs (Computer Program Components) | • AFSCP 800-7, para. 6-6i(3). | • None | 1. Permanent alphanumeric tag.<br>2. Variable revision code suffix. | Contractor | Also applies to routines and other controlled elements of a CPCI. |
| C. SOFTWARE STORAGE MEDIA | | | | | |
| Punched Cards, Magnetic Tapes, Magnetic Discs, Punched Tapes, etc. | • AFSCP 800-7, para. 6-6i(2) | • MIL-STD-100B, Chapter 400 | 15 alphanumeric characters maximum (excluding I, O, Q, S, X, and Z). | Contractor | |

Table 4-1. Summary of Government Requirements for Item Identifiers (Sheet 3 0f 3)

| Items Requiring Identifiers | Directive and Guidance Documents | Contractor Compliance Documents | Major Characteristics of Item Identifiers | Party Responsible for Assigning Item Identifiers | Comments |
|---|---|---|---|---|---|
| D. CONFIGURATION CONTROL/STATUS ACCOUNTING DATA ITEMS | | | | | |
| ECPs (Engineering Change Proposals) | • None | • DOD-STD-480A, para. 10.6 | 13 alphanumeric characters maximum, consisting of numeric identifier for basic ECP, dash, dash number for related ECPs, type (P=prel., F=formal), revision identification, (R1, R2 etc.), and correction identification (C1, C2, etc.) | Originator of ECP | |
| SCNs (Specification Change Notices) | • None | • MIL-STD-490, para. 3.3.2.3 | In numerical sequence for each specification starting with "1". | Contractor | |
| NORs (Notices of Revision) | • None | • DOD-STD-480A, para. 100.1.3. | One of the following (unless procuring activity prescribes something else):<br>1. A special series for NORs originated by the contractor.<br>2. Number of the document to be revised, plus a new revision letter. | Contractor | |
| Requests for Deviations/ Waivers | • None | • DOD-STD-480A, paras. 80.4, 90 | Same as for ECPs above. | Contractor | |
| Configuration Indexes | • None | • MIL-STD-483, para. 80.10.2(i) | Issue numbers are in numerical sequence, starting with "1". | Contractor | |

key document for both documentation and CPCIs. This supplement defines the general requirements for the USAF Computer Program Identification Numbering (CPIN) System.

The CPIN System is tied to the USAF policy of acquiring computer programs as configuration items rather than as data items. The system mostly affects AFLC and the using commands, but AFSC is required to comply with it when it is applicable.

Only two of the identifiers in Table 4-1 (the first sets for documents and CPCIs) are included in MIL-STD-482A, "Configuration Status Accounting Data Elements and Related Features."

4.5.3 Contractor Item Identifiers

Contractor systems for assigning item identifiers must satisfy all Government identification needs, as stated in the contract, as well as the contractor's own identification needs during development. The contractor's intended approach for satisfying these needs should be defined in his CM Plan.

Here are some of the factors to consider when evaluating a contractor's item identification system:

a) Document Identifiers. Do they:

   1) Meet requirements of applicable Government standards?

   2) Indicate revisions and changes?

   3) Provide for multi-volume documents?

   4) Give some clues to a document's content or usage, such as CDRL number, relation to the system documentation tree, intended operational site, or operator position?

b) Software Element Identifiers. Do they:

   1) Meet requirements of applicable Government standards?

   2) Comply with computer operating system labeling requirements?

   3) Provide both permanent base name (for retrievability) and variable tag (for identification of version)?

   4) Allow for adequate number of versions?

5) Allow for different versions of the same item to coexist and change independently?

6) Indicate the item's function?

7) Indicate the parent element or elements?

8) Provide for identification of data blocks?

c) Software Storage Media Identifiers. Do they:

1) Meet requirements of applicable Government standards?

2) Provide for identification of all software storage media to be used (e.g., decks, magnetic tapes, punched tapes, discs)?

3) Provide for proper identification of card decks used for generation and control of the software system (e.g., system generation directives, job control decks) and for updates of programs?

4) Provide for proper identification of different states of code (i.e., source code, object code, load modules, core images)?

5) Distinguish between master tapes, working tapes, and any other categories that must not be confused?

6) Include both reel numbers (for control library needs) and tape headers (to identify contents and version) for magnetic tapes?

7) Include identifiers imbedded in magnetic tapes, to be printed out upon loading?

8) Provide for identification of program listings?

d) Change Control Form Identifiers. Do they:

1) Meet requirements of applicable Government standards?

2) Provide sequential identification suitable for logging, filing, reference, and retrieval?

## 5. SPECIFIC GUIDANCE FOR CONFIGURATION CONTROL

Items, periods, and types of configuration control are discussed in this section. The types are baseline configuration control, contractor internal configuration control, and interface control. Also discussed are control libraries for software and documents.

### 5.1 ITEMS AND PERIODS OF CONFIGURATION CONTROL

Items subject to configuration control can be placed into four groups:

a) Configuration Identification Specifications. These are the System/System Segment Specifications, Development Specifications, Product Specifications, Data Base Specifications, Addendum Specifications, and Inventory Item Specifications. Much of the control effort during a development program is directed toward these documents, particularly the Development Specifications, which form the basis for CPCI design.

b) Computer Program Configuration Items (CPCIs). The primary CPCIs in a development program are application programs, but other types of programs required for use and support of the system also are subject to control. These include operating system programs, test programs, and support programs.

c) Qualification Test Documents. During qualification testing (PQT and FQT), controlling changes to the Qualification Test Procedures is as important as controlling changes to the CPCI or the Development Specification, because all evidence of satisfactory performance is based on these procedures. The Qualification Test Plan also is subject to control.

d) Technical Manuals. These include User Manuals, Positional Handbooks, and Computer Programming Manuals.

Two levels of configuration control generally are required for these four groups of items during a computer program life cycle: baseline configuration control, which applies only to baselined CPCIs and associated documentation, and contractor internal configuration control, which is applied to software and documents during the development process, before the items are baselined. Figure 5-1 shows the periods during which these two levels of control generally are applied.

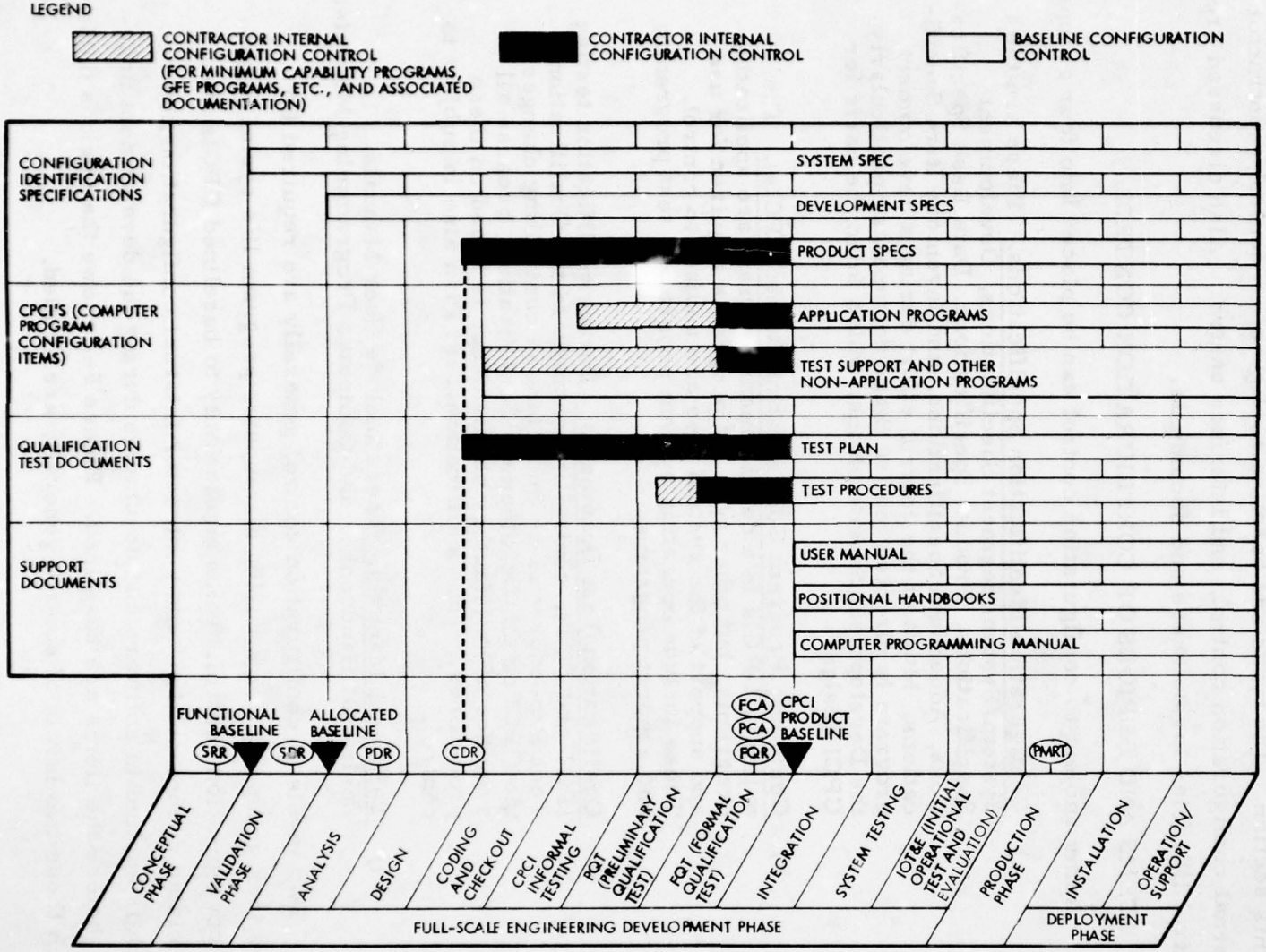



Figure 5-1. Items and Periods of Configuration Control

For contractor internal control, control before FQT is selectively applied to routines that have been intentionally developed early or that are supplied by the computer system manufacturer or by the Government, and then is totally applied to the entire CPCI for FQT testing. Product Specifications and Test Plans usually are subject to contractor internal configuration control from CDR on.

Some of the effects of baseline and contractor internal controls on the development process can be seen in Figure 5-2.

## 5.2 BASELINE CONFIGURATION CONTROL

Baseline configuration control is the control that the procuring activity imposes on configuration items and identification documents after they have met contractual requirements. This subsection discusses the following aspects of baseline configuration control:

a) Types of changes

b) Forms

c) Configuration Control Boards (CCBs)

d) Evaluating change proposals

e) Procedures for control

f) Maintenance of baseline documents

g) Maintenance of baseline software items

h) System turnover and PMRT

### 5.2.1 Types of Baseline Changes

Proposed changes to a baselined CPCI or document take two general forms: engineering changes, which are changes to an established baseline, and deviations or waivers, which are departures from an established baseline that do not change the baseline.

#### 5.2.1.1 Engineering Changes

An engineering change is a change in the configuration of a CPCI after the configuration identification has been established as a baseline. If a proposed engineering change appears to affect the Government's interest, it is considered a Class I change and the contractor must

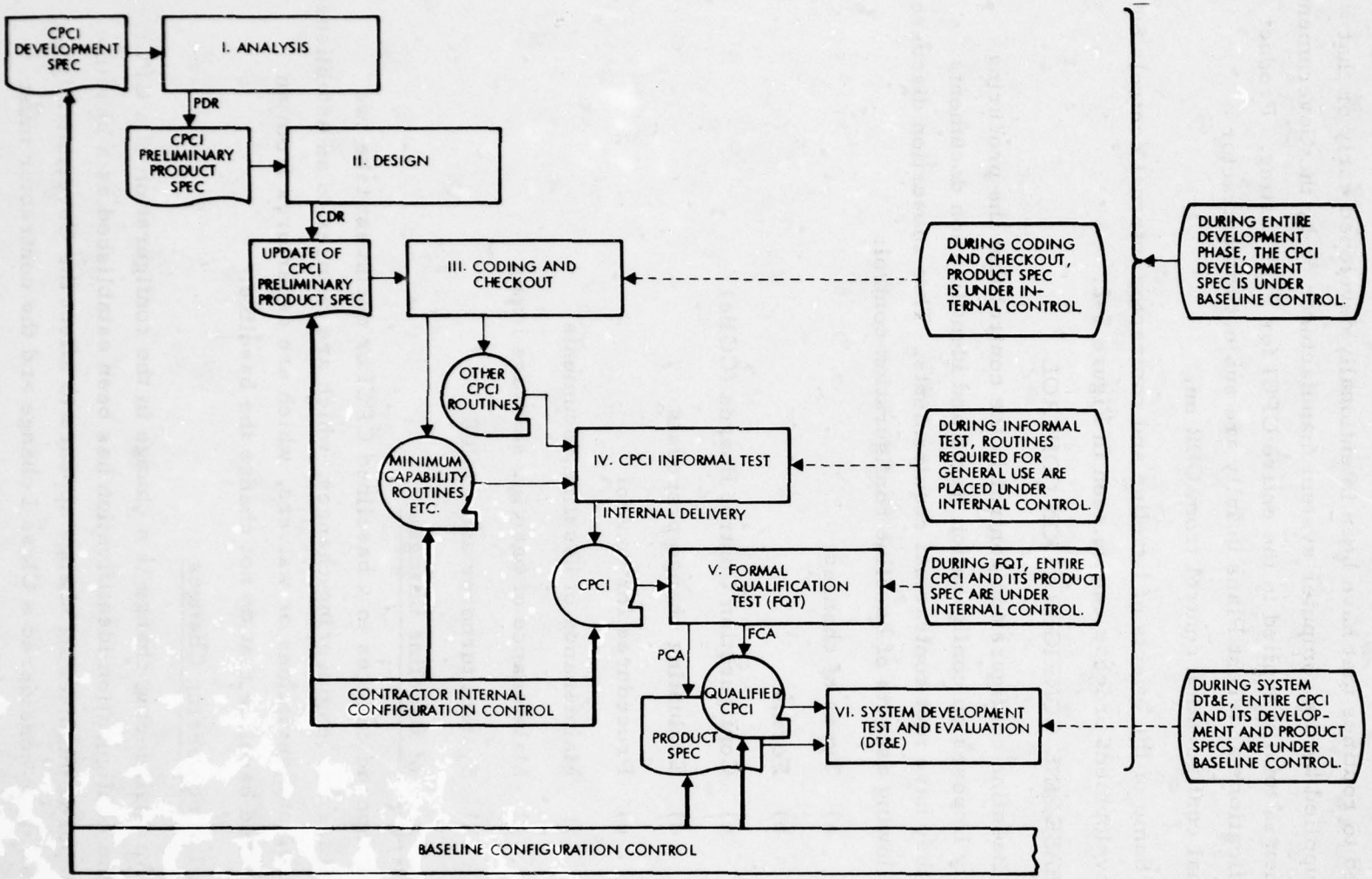



Figure 5-2. Effects of Baseline and Contractor Internal Control Systems on Development Process

submit it to the procuring activity for approval before implementing it. If the change does not appear to affect the Government's interest, it is a Class II change. On most contracts, the contractor is permitted to implement a Class II change without approval but must report it to the procuring activity for concurrence that the classification is correct.

More specific criteria for these classes are as follows (paraphrased from DOD-STD-480A, paragraphs 4.2.1 and 4.2.2, and other DOD documents):

a) Class I Change. An engineering change is in Class I when it affects one or more of these factors:

1) Functional configuration identification (i.e., normally the System or System Segment Specifications).

2) Allocated configuration identification (i.e., Development Specification).

3) Product configuration identification (i.e., Product Specification, but not including program listings or actual data values, which are covered in item (4)).

4) The following technical requirements defined in the product configuration identification (i.e., Product Specification, including program listings and actual data values):

(a) Performance outside stated tolerances.

(b) External interface characteristics.

5) Nontechnical contractual provisions: fee, incentives, cost, schedules, or guarantees or deliveries. (However, changes that apply only to nontechnical provisions of a contract should not be processed as engineering changes.)

6) Other factors: Government-furnished equipment (GFE); safety; operational, test, or maintenance computer programs; compatibility with support equipment, trainers, or training devices or equipment; configuration to the extent that retrofit action would be taken; delivered operation and maintenance manuals for which existing contracts do not provide adequate change/revision funding; preset adjustments or schedules affecting operating limits or performance to such an extent that new identification numbers must be assigned:

skills, manning, training, biomedical factors, or human engineering design.

Class I changes should be limited to those that are necessary or offer significant benefit to the Government. They may or may not require contract costs to be increased. A compatibility change, for example, is a Class I no-cost change to correct a deficiency that keeps an item from working properly.

b) Class II Changes. An engineering change is in Class II when it does not affect any of the Class I factors listed above. Examples of Class II changes are correction of spelling or typographical errors, addition of clarifying comments, program changes that do not affect external interfaces or degrade performance, recompilations within specified limits, and changes to adaptation data. The procuring activity should clearly define any special kinds of changes it wishes to include in Class II. The Government considers that a Class II change is justified if it benefits the originating contractor and is not detrimental to the Government. All Class II changes must be within scope of the contract.

The above discussion of Class I and II changes is based largely on DOD-STD-480A because it is the most recent DOD definition of these classes. Factors clearly applicable only to hardware have been omitted, and some statements based on other DOD documents have been included. For contractual applications, refer directly to DOD-STD-480A, MIL-STD-483, or other applicable documents.

5.2.1.2 Deviations and Waivers

Deviations and waivers give a contractor relief from the technical requirements of a baseline for specific instances when permanent changes are not acceptable. A deviation is a written authorization granted prior to a development activity to permit a contractor to depart from a particular performance or design requirement for a specified product or period of time. A waiver is a written authorization to deliver a CPCI or hardware CI that has been found to depart from requirements after a development activity but that is still suitable for use or rework. Waivers should not be granted unless there is an overriding benefit to the Government.

### 5.2.2 Baseline Configuration Control Forms

The following kinds of standard reporting forms are used in baseline configuration control:

a) Engineering Change Proposal (ECP). ECPs are used by contractors and Government activities to propose engineering changes to configuration items. (See Table 3-1 for ECP description and RSS references.)

b) Specification Change Notice (SCN). SCNs are used by contractors to propose, transmit, and record specification changes. (See Table 3-1.)

c) Advance Change/Study Notice (ACSN). ACSNs are used by contractors to provide the procuring activity with advance information about proposed routine changes. (See Table 3-1.)

d) Notice of Revision (NOR). An NOR is used by the contractor who originates an ECP to transmit changes required in another contractor's document because of the ECP. An NOR may be used for CPCI document changes only if the CPCI is an off-the-shelf item. (See Table 3-1.)

e) Request for Deviation/Waiver. A contractor uses this form to request and document a temporary departure from baseline requirements, either before building the item (deviation) or after (waiver). (See Table 3-1.)

f) Contract Document Change Notice (CDCN). CDCNs are used by contractors to propose, transmit, and record changes to contractual documents other than specifications, such as test plans and procedures, CM plans, SOWs, CDRL pages, etc. CDCNs are prepared on AFSC Form 417. (See AFSCP 800-7, paragraphs 1-17d and 3-3c.)

g) Task Change Proposal (TCP) and Contract Change Proposal (CCP). TCPs and CCPs are used by a contractor to propose non-technical contract changes. (See AFSCP 800-7, paragraphs 3-3.a(2) and 3.3.c and Contract Change Proposal entry in Attachment 2.)

h) Configuration Control Board Directive (CCBD). CCBDs are used by the program Configuration Control Board (CCB) to document all decisions and direction concerning change proposals. AFSC Forms 318, 318a, and 318b are used for this purpose. (See AFR 65-3, paragraph 3-6d, and AFSCP 800-7, paragraphs 1-6, 1-17b, and 3-4b.)

i) Time Compliance Technical Order (TCTO). TCTOs are used by AFSC and AFLC after system turnover to authorize the using command to use new or modified/changed computer programs. (See AFR 800-14, Volume II, paragraph 6-6d; AFSCP 800-7, paragraphs 3-10 and 6-60; MIL-T-38804 (USAF); T.O. 00-5-15; and AFLCM 66-14. Also see item C of Comments under MIL-T-38804 in Table 2-1, Sheet 2.)

j) Deficiency Reports. Deficiency reports are used by the using command to report deficiencies in computer programs or related documents following system turnover, in accordance with the O/S CMP. Deficiency reports are classified as emergency or routine. (See AFR 800-14, Volume II, paragraph 10-7.)

k) Change Reports. Change reports are used by the using command to propose changes to computer programs or related documents following turnover, in accordance with the O/S CMP. (See AFR 800-14, Volume II, paragraph 10-8.) One such form, called a "Computer Program Configuration Sub-Board Item Record" (AFLC Form 75), is outlined in AFR 800-14, Volume II, Attachment 3.

The forms listed above are all Government forms. Those that contractors are required to submit as part of baseline configuration control should be listed in the contract CDRL, with references to appropriate DIDs (Data Item Descriptions) or AFSC or DD forms.

Additional types of forms are used by contractors to document baseline configuration control problems and changes. These forms generally are the same ones used in the contractor's internal configuration control system and are discussed in subsection 5.3.2 of this guidebook.

### 5.2.3 Baseline Configuration Control Boards (CCBs)

Configuration Control Boards (CCBs) are responsible for change decisions. Prior to PMRT, procuring activities and contractors both require CCBs, and after PMRT, an Air Logistic Center CCB usually performs this function for the entire system.

#### 5.2.3.1 Procuring Activity CCBs

The program office should establish a system-level CCB during the validation phase to evaluate and approve or disapprove proposed changes to baselines throughout the acquisition period. All ECPs and major or critical deviations and waivers are subject to review by this CCB.

The program manager usually is chairman of this CCB. Its members normally include the top managers of each functional area in the program office and representatives from AFLC, using commands, other participating Government agencies, and the ICWG. Specialists from engineering groups, not-for-profit contractors, and similar organizations may participate in CCB activities as advisors. Contractors are not members of the CCB, but may attend meetings on request.

Final approval/disapproval authority in the CCB rests with the chairman; other members advise and recommend.

CCB procedures usually are prepared and maintained by the CMO, who also functions as the CCB secretariat. Procedures should cover preparation of agendas, conduct of meetings, and maintenance of records. Each change decision should be documented on a Configuration Control Board Directive (CCBD), which is directive on all personnel who must act on a change. Figure 2 in AFR 65-3 lists a typical sequence of steps in processing ECPs.

In addition to controlling baselines, the CCB is the appropriate agency to formally establish each baseline. (See AFSCP 800-7, paragraph 3-6.)

The procuring activity CCB is responsible for both hardware and software elements of the system. If the system is large and complex, processing of change proposals will be facilitated by formation of a separate software CCB such as the Computer Program Configuration Sub-Board (CPCSB) described in subsection 5.2.3.3. The screening function described there for preliminary review of ECPs also is applicable to procuring activity CCBs.

Procuring activity CCBs usually cease to function after PMRT.

5.2.3.2 Contractor CCBs

A contractor CCB performs about the same functions for the contractor's products that the program office CCB performs for the entire system. Its primary responsibilities are to approve or disapprove Class I change proposals and to concur or not concur with the classification of Class II changes.

The chairman of a contractor CCB usually is the project manager or system engineer, and other members are the development managers, the system engineer (if he is not chairman), the configuration manager, the quality assurance manager, and key technical personnel. The planning and control manager also may participate when cost or schedule are affected. As in other CCBs, the chairman has final authority for all decisions.

Contractor CCBs meet weekly or at other regular periods, and also whenever the chairman believes it necessary. Specific attendance at CCB meetings usually depends on the areas impacted by the change proposals listed on the agenda.

The contractor configuration manager usually organizes a CCB soon after start of the contract and prepares and maintains the procedures for CCB operation. The configuration manager or his representative is the CCB secretariat, who prepares meeting agendas, review materials, and meeting minutes.

Contractors sometimes have technical staff personnel screen Class I change proposals to perform preliminary evaluation and to make recommendations to the CCB.

5.2.3.3 Deployment CCBs

A post-PMRT CCB is formed by the supporting activity (usually an Air Logistic Center) in accordance with the system O/S CMP. This CCB is responsible for all changes to the system and its configuration items during the remainder of the system life-cycle.

Membership of this CCB consists of representatives of all involved agencies and system functional areas, including CM, programming, engineering, and test.

A Computer Program Configuration Sub-Board (CPCSB) sometimes is formed to facilitate processing of computer program changes. Board members usually represent the supporting command, using commands, and appropriate programming and engineering personnel. An additional screening function also may be created to support the CPCSB or the CCB itself. Responsibilities of the CPCSB and screening function are discussed in AFR 800-14, Volume II, paragraph 6-11.

### 5.2.4 Evaluating Baseline Change Proposals

Careful evaluation of all ECPs and requests for deviations and waivers is required to ensure that changes are limited to those that are necessary or offer significant benefit to the Government. Necessary or beneficial changes, according to AFR 65-3, paragraph 3-2, are those that meet any of the following criteria:

a) Correct deficiencies.

b) Satisfy changes in operational or logistic support requirements.

c) Produce substantial life-cycle cost savings.

d) Prevent or allow desired slippage in an approved schedule.

More specific considerations are mentioned in AFR 65-3, paragraphs 3-7, and AFSCP 800-7, Figure 3-3. Those pertaining to CPCI changes are the following:

a) Effect of the change on CPCI performance.

b) Effect on overall system performance and compatibility requirements.

c) Effect on other CPCIs or hardware CIs in the system.

d) Effect on CPCIs or hardware CIs in other systems.

e) Effect on the CPCI's external interfaces.

f) Effect on the CPCI's documentation, including configuration identification specifications, test procedures, user manual, positional handbook or operator manual, and computer programming manual.

g) Adequacy of the change proposal for translation into detail design changes.

h) Mathematical, scientific, or data processing aspects of the change, including consideration of other, similar development efforts in process.

i) Amount of testing required on the CPCI itself and on any other elements of the system to validate implementation of the change, and cost and schedule impact of such testing.

j) Effect on multiple-location applications.

k) Effect on nuclear safety.

l) Effect on contract cost.

m) Effect on schedule.

Detailed analysis or investigation of change proposals to define consequences in the areas listed above should be performed by individual members of the CCB or other knowledgeable persons prior to CCB review. This preliminary examination may be assigned to program office functional areas or to participating Contract Administrative Services (CAS), such as AFPRO or DCASR. (See paragraph 3-7c of AFSC Supplement 1 to AFR 65-3.)

A more routine kind of preliminary review of change proposals to ensure proper preparation of the forms should be performed by the CMO as part of its CCB secretariat responsibilities. Items to be checked include those listed in AFR 800-3, paragraph 9-24d.

5.2.5 Baseline Configuration Control Procedures

After a configuration identification specification or CPCI is baselined, it can be changed only through the baseline configuration control procedures defined in the system CM documents.

Major factors to consider in preparing or evaluating a procedure for baseline configuration control include the following:

a) Reporting Problems and Proposing Changes. Most change proposals arise from the recognition of problems, errors, or deficiencies, the rest from new requirements. Reporting problems and proposing changes are indispensable parts of the development process and should be encouraged by keeping forms and procedures as simple and convenient as possible.

b) Evaluating Change Proposals. Class I change proposals should be thoroughly evaluated by the procuring activity CCB before any decision is made. Some guidelines for this have already been discussed in paragraph 5.2.4.

c) CCB Actions. Possible actions of the procuring activity CCB after evaluation of a change proposal include the following: (1) approval, (2) approval with changes, (3) disapproval, (4) deferral of action for further study, with specific due date, (5) referral to higher authority with recommendation, when approval is outside the authority of the chairman.

d) Approval Principles. A list of important principles governing the approval or disapproval of ECPs is given in AFR 65-3, paragraph 3-7h (2).

e) Configuration Control Board Directives (CCBDs). The CCBD is the CCB's official notification of its decision on a particular change proposal. The CMO is responsible for distribution of CCBDs to Government agencies required to take action. Contractors may receive copies of the CCBD comments, but only through the Procuring Contracting Officer (PCO) and only for information.

f) Contract Authorization. An approved CCBD directs the PCO to issue a contract order or supplemental agreement to the contractor, authorizing implementation of the ECP.

g) Processing Time Spans. Timely implementation of worthwhile changes is a key requirement of configuration control. This requires establishment of maximum time spans for change processing, assignment of priorities to ECPs by the originator, and use of followup controls. (This subject is discussed in AFSCP 800-7, paragraphs 1-6, 3-3, and 3-5; AFR 65-3, paragraphs 3-7f and 3-7g (including AFSC Supplement 1 comments); DOD-STD-480A, paragraphs 4.5, 4.8.9, 4.9.4, and 4.9.5; DODD 5010.19, paragraphs V.C.3 and V.C.5; and DODI 5010.21, paragraph VI.D).

h) Contingency Procedures. The baseline configuration control process should include provisions for accelerating the normal change cycle when necessary to prevent serious delays or other undesireable consequences. Instead of weeks or days, there may be times when hours are the measure of the required implementation time. During system tests, changes sometimes are processed in as little as one hour, from problem occurrence through analysis, design, coding, test, approval, and installation. Any formal paperwork omitted in the heat of such a situation should be completed as soon after as possible and processed through regular channels to keep the records straight.

i) Change Tracking. CMO records should include a log that provides the current status of all change proposals being processed through the program office. This log should permit each change to be tracked from initiation through receipt, CCB approval, contract authorization, and implementation.

The principal Government standard to consider when developing and managing a baseline configuration control system is DOD-STD-480A, as modified by MIL-STD-483, Appendix XIV.

The CMO should document baseline configuration control procedures in sufficient detail to guide CMO personnel and all other program office personnel who will be involved in the process. Adjustments to the procedures usually are required at intervals to improve efficiency or to meet unforeseen problems.

An essential part of developing a coherent and workable configuration control procedure is charting the basic flow of events in the problem/change cycle. Figures 5-3 through 5-5 are examples of such charts for three different control situations: Class I changes to baseline specifications, Class II changes to baseline specifications, and Class I/II changes to baseline specifications and CPCIs during Development Test and Evaluation (DT&E) at a system test site.

These charts do not include the processing of change impacts on other baseline specifications, on baseline CPCIs, and on support documents, except that the third chart treats CPCI and document changes as a coordinated package. Interfaces with hardware change control also are omitted.

The first chart in the series (Figure 5-3) shows the sequence of control activities for Class I in-scope (compatibility) and out-of-scope (design) changes to baseline specifications. This sequence applies to Class I changes to the System or System Segment Specification and to the Development Specifications during most of the full-scale engineering development phase and to the Product Specifications after the CPCI product baselines but before system DT&E.

The second chart (Figure 5-4) shows the sequence of control activities for Class II changes to baseline specifications. For Class II changes, only page 1 of the ECP form is required, or the contractor's own form may be used if it satisfies the requirements of DOD-STD-480A, paragraph 4.6.2. The Document Update Transmittal (DUT) form described

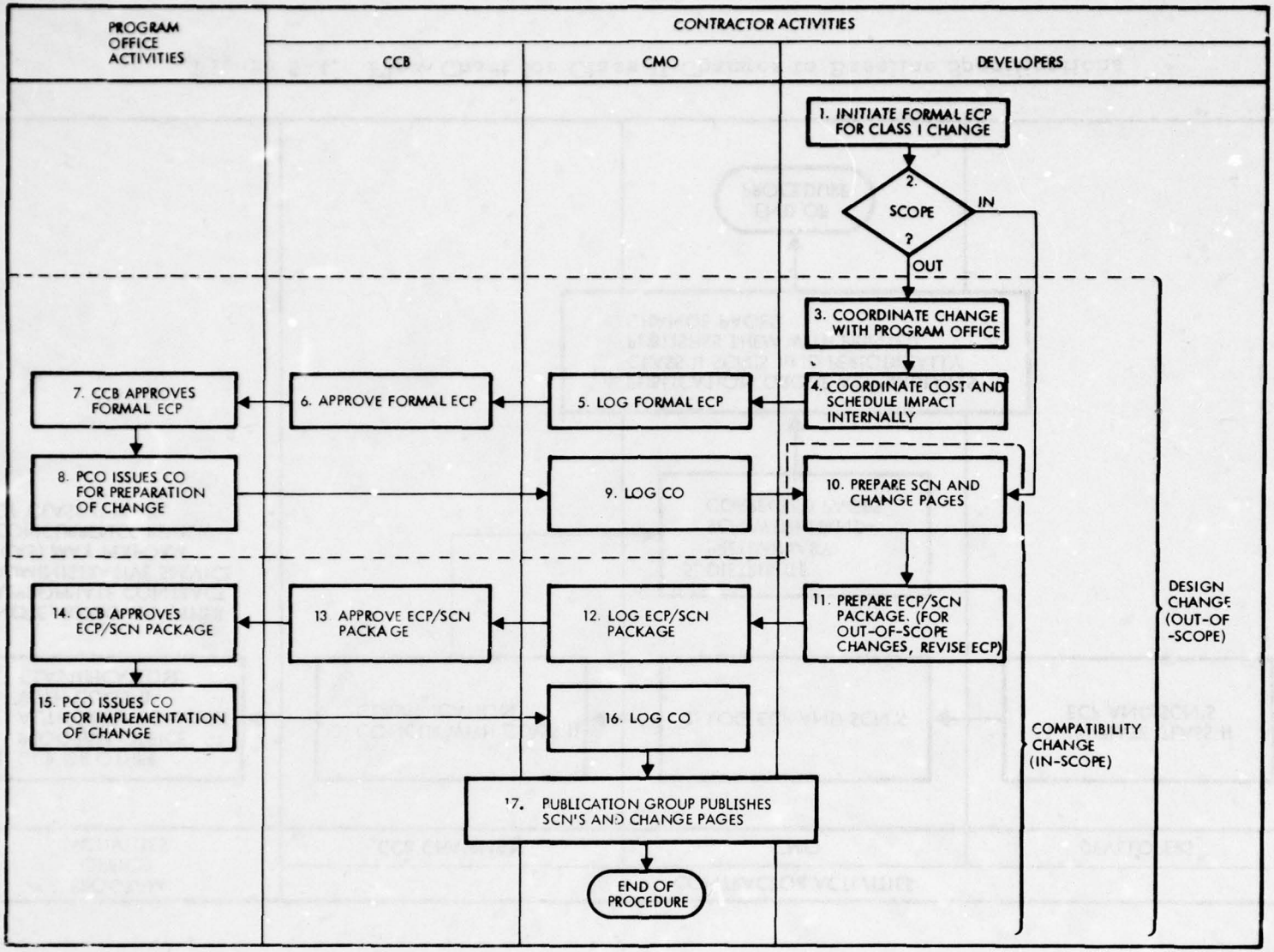



Figure 5-3. Flow Chart for Class I Changes to Baseline Specifications

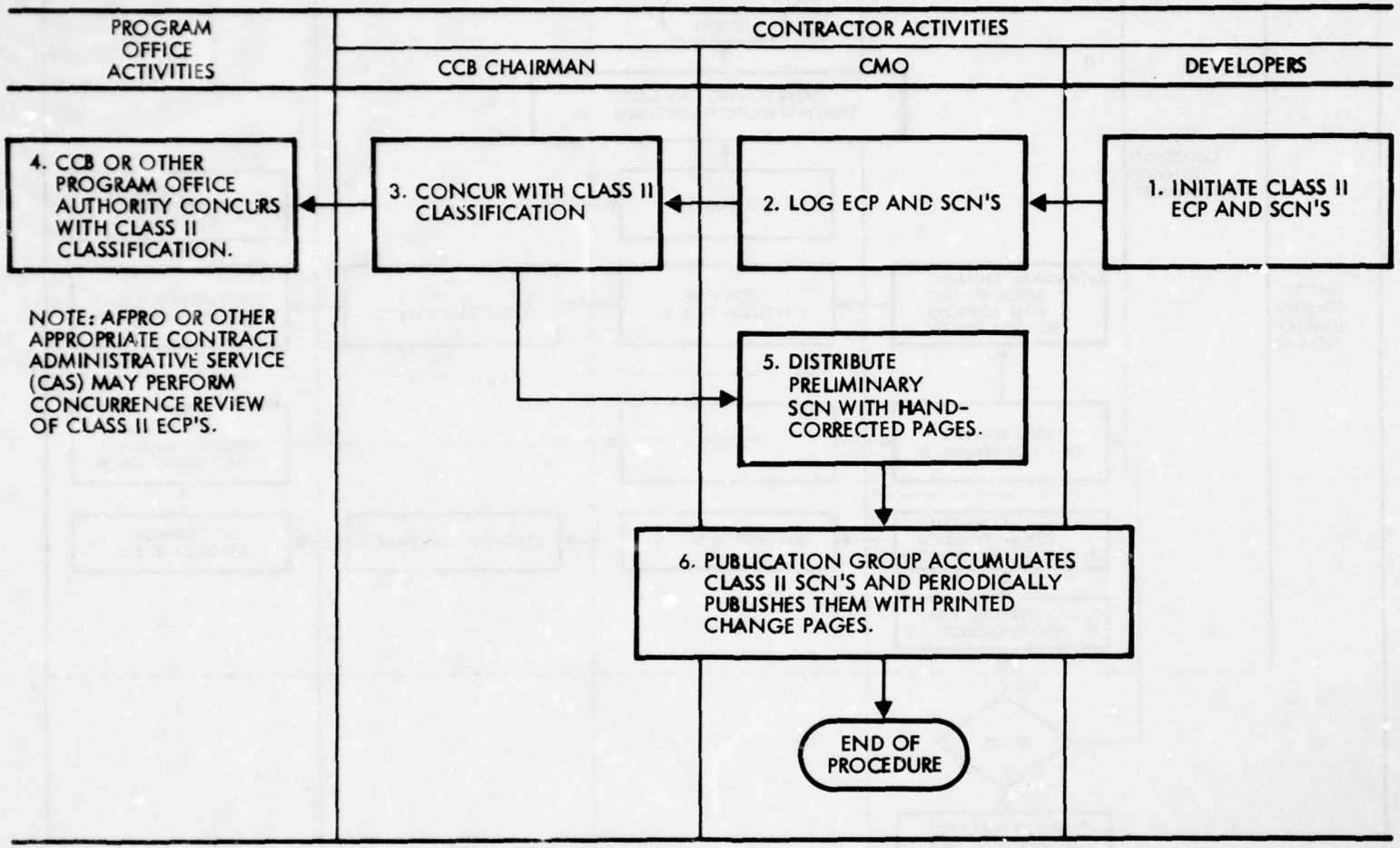



Figure 5-4. Flow Chart for Class II Changes to Baseline Specifications

AD-A076 542 TRW DEFENSE AND SPACE SYSTEMS GROUP REDONDO BEACH CA F/G 5/1
AIRBORNE SYSTEMS SOFTWARE ACQUISITION ENGINEERING GUIDEBOOK FOR--ETC(U)
NOV 78 E JUCEVIC F33657-76-C-0677
UNCLASSIFIED TRW-30323-6010-TU-00 ASD-TR-79-5024 NL

2 OF 2
AD A
076542

END
DATE
FILMED
6-80
DTIC

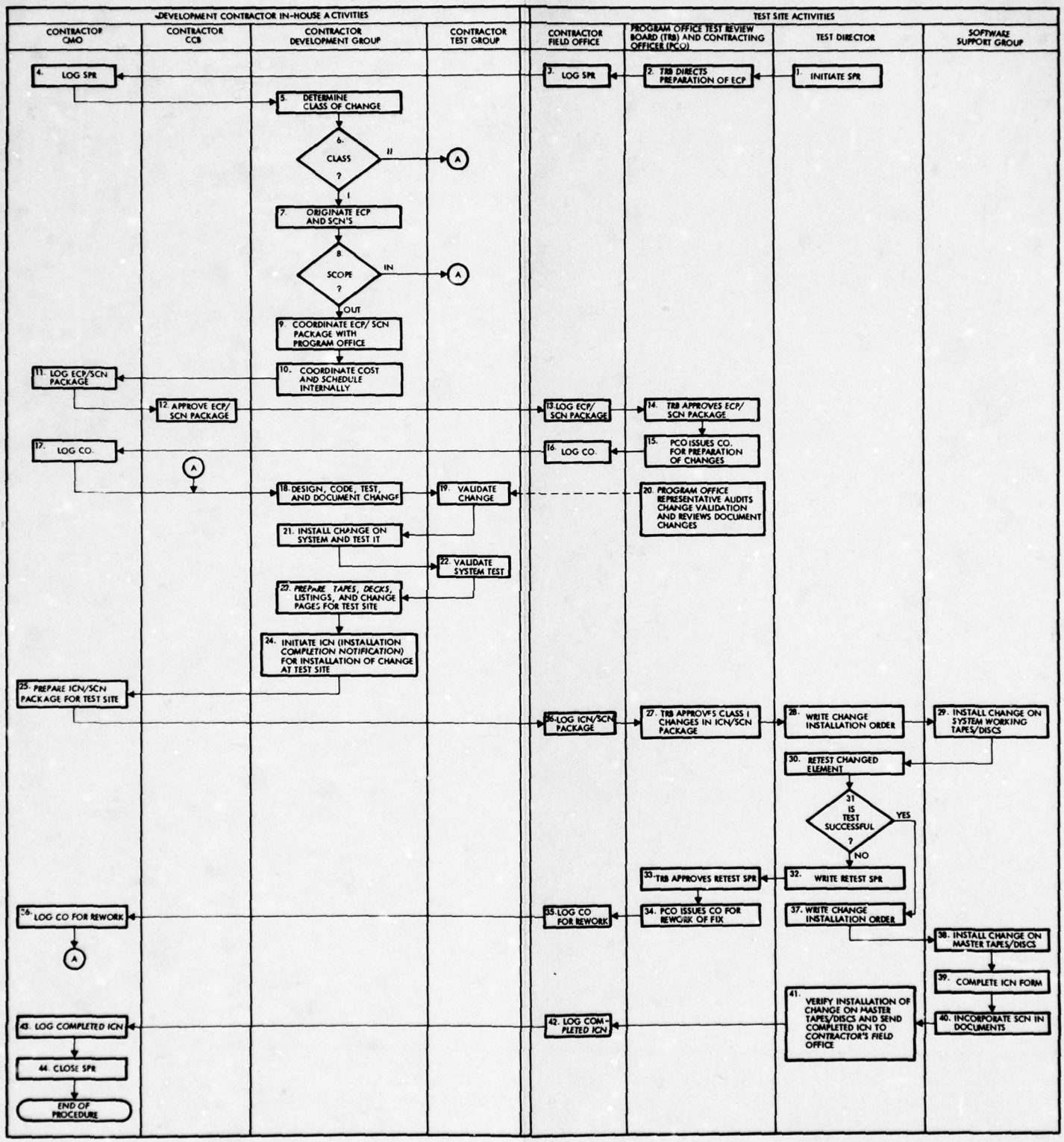



Figure 5-5. Flow Chart for Class I/II Changes to Baseline Specifications and CPCIs During DT&E

under "Internal Configuration Control Forms" (paragraph 5.3.2) in this guidebook has been used for this purpose. The DUT also may be used as a substitute for the SCN up to the time the change pages are distributed.

Normally, contractors are required to submit batches of Class II changes at specified intervals for concurrence review, instead of submitting them individually. Some program offices permit Class II changes only when the affected pages also contain Class I changes.

As stated earlier, the major difference between Class I and Class II change processing is that the Government requires approval of Class I changes prior to implementation but only requires concurrence with the classification of Class II changes. When concurrence is not given for a Class II change, the contractor must resubmit it as a Class I change or drop it.

Another difference between Class I and Class II change processing is that Class II changes must not impact other baseline items. Any changes having such impact must be processed as Class I changes. Still another difference is that Class II changes must be within the scope of the current contract.

The third chart (Figure 5-5) shows the sequence of control activities for both Class I and Class II changes during system DT&E (Development Test and Evaluation) at a test site. Activities at the test site and at the development contractor's home facility are both shown. The time lag resulting from separation of the two facilities creates additional coordination and tracking requirements for the CMO. The program office review board in this chart is called the Test Review Board (TRB), but its function is that of a CCB.

Configuration control procedures for IOT&E (Initial Operational Test and Evaluation) and later periods are not covered in this guidebook. The same concepts are applicable but their implementation is increasingly dependent on the policies and practices of the other commands involved. The procuring activity CCB continues to be responsible for control of established system baselines until PMRT.

### 5.2.6 Maintenance of Baseline Documentation

Contractors must have efficient methods for implementing documentation changes resulting from the configuration control process. Baseline specifications and impacted test and support documents must be kept up-to-date through regular updates and periodic revisions, in compliance with the requirements of MIL-STD-490, paragraph 3.3, and MIL-STD-483, Appendix VIII. Change pages to specifications are issued with SCNs (Specification Change Notices), and change pages to support documents with CDCNs (Contract Document Change Notices; see paragraph 5.2.2f of this guidebook).

A satisfactory documentation maintenance system will meet part of the Government's requirement that software development contractors possess the equivalent of a hardware engineering release system. (See AFR 800-4, Volume II, paragraph 6-6c; AFSCP 800-7, paragraph 6-6j; and MIL-STD-483, paragraph 100.3, for requirement statements.) An engineering release system is a method for formally issuing specifications, drawings, and other engineering data to manufacturing, procurement, etc., to use in development and production of a hardware configuration item. In software development, there appear to be two activities sharing this function: the documentation maintenance activity discussed above and the software maintenance activity discussed in the next subsection.

### 5.2.7 Maintenance of Baseline Software

Software assembly and maintenance is an integral part of the configuration control process, even though it may be performed by another contractor area besides the CMO. Proper performance of this function is essential to the continuing control and traceability of coded program configurations. Types of tasks performed by software assembly and maintenance personnel include the following:

a) Assembly, or construction, of the master library of source and object code for all programs under configuration control.

b) Updating of the master program library in accordance with instructions from the CMO.

c) Control of the master program library tapes and/or disks.

d) Provision of working copies of the master program library to project design and test personnel.

e) Maintaining library update records consisting of the card updates, a copy on tape of each update, a complete source listing of each routine, and the job control cards.

f) Generating and maintaining master tapes and update records for the development facility (or operational support facility) computer operating system in a manner similar to the above.

g) Verifying that all approved changes have been incorporated in master tapes and decks in accordance with the authorizing change control documents. (Quality Assurance may perform this task.)

### 5.2.8 Turnover and Transfer

All program office responsibilities for management of a system, including CM, are shifted to the supporting command at an event called the Program Management Responsibility Transfer (PMRT), which should occur early in the production phase. Usually preceding that is the formal turnover of the system hardware and software to the using command. For computer resources, including computer programs, AFR 800-4 governs the PMRT and AFR 800-19 governs the system turnover. AFR 800-14, Volume II, also discusses turnover and transfer in Chapter 9.

Planning for these two events begins early in the acquisition cycle during preparation of the PMD, PMP, CRISP, and O/S CMP. A PMRT agreement and a turnover agreement subsequently are prepared to govern the actual events. In addition, a turnover certificate documents deficiencies and corrective action responsibilities and forecast dates.

## 5.3 CONTRACTOR INTERNAL CONFIGURATION CONTROL

Contractor internal configuration control is the control the development contractor imposes on non-baselined CPCI computer programs, data, and documents during the development process. This control process and its major products are illustrated in Figure 5-6. The following aspects of this process are discussed in this subsection:

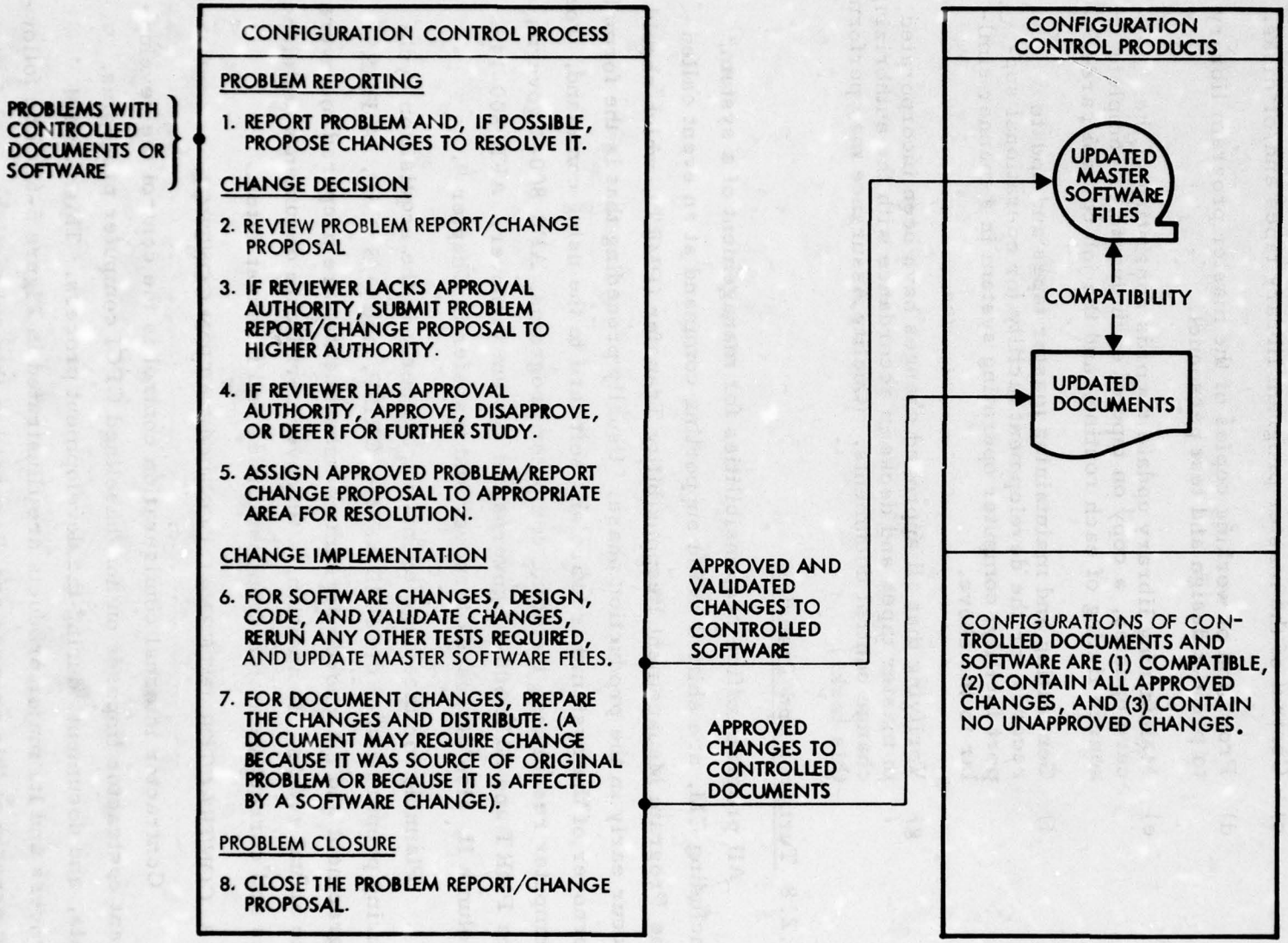



Figure 5-6. Contractor Internal Configuration Control Process and Products

a) Types of internal configuration control changes

b) Internal configuration control forms

c) Change approval authority

d) Procedures for internal configuration control

e) Maintenance of internally controlled documentation

f) Maintenance of internally controlled software

### 5.3.1 Types of Internal Configuration Control Changes

Contractor internal configuration control changes may or may not be classified into different types. A division into at least major and minor types usually aids evaluation and speeds up processing because most changes are minor. Here is one way to establish such a two-way division:

a) Major Change

1) Affects the approved code-to version of the Product Specification

2) Affects internal interfaces

3) Affects internal schedules

b) Minor Change

1) A design error that does not affect the code-to Product Specification, internal interfaces, or internal schedules.

2) A minor coding error

3) An editorial correction

Instead of, or in addition to, such a division of changes, a contractor may choose to consider the importance of the software items to the success of system operation. A minor change to a mission-critical module might warrant a higher change classification than a major change to a utility module, for example.

When a proposed contractor internal change affects a baselined item, the contractor must submit the appropriate class of ECP (I or II) for the baseline change to the procuring activity CCB. If the baseline change is Class II, both the internal change and the baseline change may be implemented as soon as the ECP is sent to the procuring activity CCB. If the baseline change is Class I, however, both internal and baseline changes must await ECP approval by the procuring activity CCB. Such internal changes are called "impact" changes.

Approval authorization levels for internal changes are discussed in subsection 5.3.3.

### 5.3.2 Internal Configuration Control Forms

Many different kinds of forms are used by software development contractors for internal configuration control purposes. Most of them are either problem reports or change orders. The problem reports often contain a change proposal or change recommendation section. Separate forms generally are used for software and for documentation. Problem reports and change orders for hardware items also must be considered because of their interactions with the software change control process.

Some of the forms of these various types that are used by software development contractors are described in Table 5-1. Copies of a coordinated set of the first five forms in this table (DPR, DUT, SPR, SMR, and DBCR) are contained in Appendix D of this guidebook.

In terms of forms, the general configuration control process can be considered to have six steps:

a) Problem report.

b) Problem analysis.

c) Problem solution.

d) Change order.

e) Change implementation.

f) Closure of problem report.

Two examples of this process for different control purposes are shown in Figure 5-7. In addition to the steps shown, each newly generated problem report and change order is logged by the CMC and reviewed by the CCB or other review authority before it proceeds to the next step.

### 5.3.3 Internal Configuration Control Change Approval Authority

Contractors usually do not employ a formal CCB for internal configuration control. Instead, proposed major changes require approval by

## Table 5-1. Types of Contractor Configuration Control Forms

| Form Type | Name of Form | Purpose of Form |
|---|---|---|
| For Documentation Problems and Changes | DPR (Design Problem Report) | Used by Government and contractor personnel to report problems with specifications and other technical documents during design reviews, configuration audits, and other occasions when documents are subject to review. |
| | DUT (Document Update Transmittal) | A cover sheet for distributing handcorrected pages containing approved changes prior to formal publication. Lists attached pages and references related DPRs or other control documents. |
| For Software Problems and Changes | SPR (Software Problem Report) | Used to report a known or suspected deficiency in software. During software tests, may be used to report all problems, whether problem is believed to be caused by the software under test, by the test procedures, by the computer hardware or operating system, or by other elements of the system. |
| | SMR (Software Modification Record) | Used in one of the following ways: (1) to identify and describe computer program changes required to correct a problem reported in an SPR, (2) to identify and describe computer program changes required to implement an ECP, (3) to accompany a Data Base Change Request (DBCR) that identifies and describes data base changes requested to correct an SPR problem or to implement an ECP, or (4) to close an SPR when no changes of any kind are required or approved. An SMR must be prepared to answer every SPR that is opened. An SMR also is used by programmers to release a new computer program to the CMO or software control library. |
| | DBCR (Data Base Change Request) | Used to request and implement changes to a data base. |
| For System, Hardware, or Software Problems and Changes | DR (Discrepancy Report) | Used to report any problem or discrepancy occurring during integrated system test or during system operation |
| | ECR (Engineering Change Request) | Used to initiate drawing or specification changes to a hardware item in response to a DR or an approved ECP. |
| | EO (Engineering Order) | Used to implement changes to a hardware or software item. |
| Miscellaneous | PR (Problem Report) or TR (Trouble Report) | Alternate names for DRs, DPRs, or SPRs |
| | SCR (Software Change Request) | If software problems and software changes are recorded and tracked by two different groups in the project (e.g., Product Assurance and CMO), the SPR is used for reporting problems and the SCR for proposing changes. An SMR still is required. |
| | CCR (COMPOOL Change Request) | For requesting COMPOOL changes when JOVIAL is the programming language used. |
| | SPMR (Software Problem/ Modification Report | A combination of the SPR and SMR for small development programs. Saves paper work but may be inconvenient when a single change corrects many problems or vice versa. |
| | CAR (Correction and Release) Form | Used for documenting a software correction and releasing it to another organization, either internally (from developers to testers or to software library) or to the procuring activity or another contractor. |

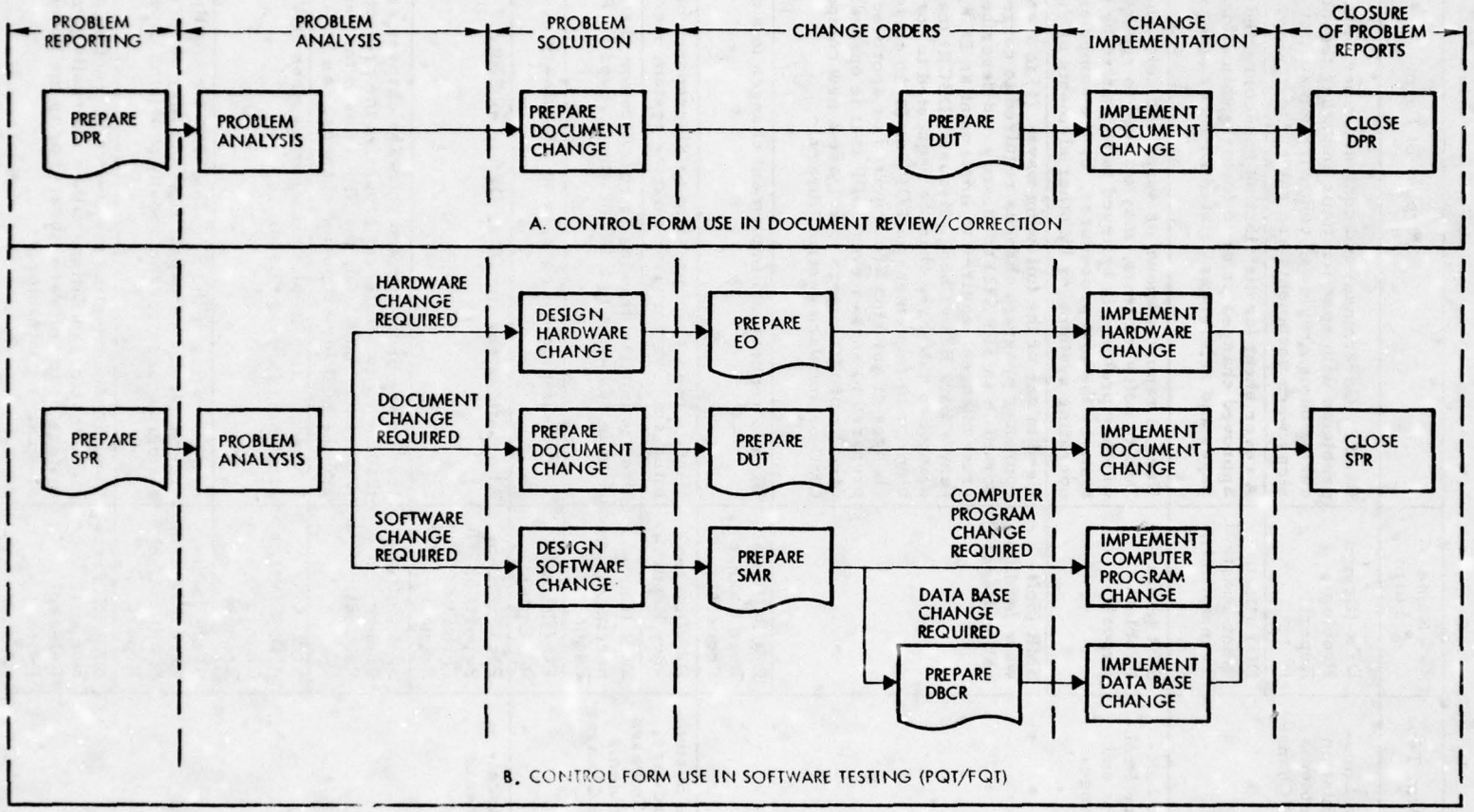



Figure 5-7. Examples of Configuration Control Form Use

several high level technical personnel such as the system engineer and the development manager. Minor changes may require only a quick review by the system engineer to ensure that they have not been under-classified.

The procuring activity normally has no contractual authority over changes to items that are not part of the formal configuration baseline and are under only internal control. Visibility of this change process and its results is achieved through formal design reviews, informal technical interchange meetings, and witnessing PQT and FQT tests. In addition, the procuring activity should be on distribution for problem reports, change orders, and internal status accounting reports.

### 5.3.4 Internal Configuration Control Procedures

Some of the important characteristics of an effective internal configuration control procedure are the following:

a) Control Applied to Appropriate Items at Appropriate Points. Configuration control should be applied to an item at those times when changes must be formally limited to necessary ones or when users of the item other than its creators need to know what is happening to it. Any other control probably is not needed and may interfere with development activities.

b) Appropriate Degree of Control. Forms, records, approval authority, and processing steps should be the minimum required to provide reasonable assurance that configuration integrity will be maintained.

c) Responsiveness to Schedule Needs. Response times for changes must be compatible with their need in the current development period.

d) Suitable Verification Techniques. Checksum programs or other verification methods should be employed to ensure that controlled programs have not undergone unauthorized or inadvertent changes.

The principal periods of internal configuration control were briefly mentioned at the beginning of this section and are shown in the bar chart in Figure 5-1. From CDR to the start of FQT, the following items usually are subject to internal control:

a) CPCI Preliminary Product Specification.

b) Minimum capability application routines being placed on the master library for general use.

c) Compiler and assembler. If these or other portions of the operating system are under development, a plan for incremental release should be developed to ensure items will be available when needed.

d) Skeleton executives, utility routines, and required data base files.

During FQT, the following items should be under internal control as parts of the FQT Test package:

a) CPCI updated Product Specification, with listings.

b) All CPCI cards, tapes, disks, and related items.

c) The entire computer operating system.

d) Diagnostic programs for hardware checkout.

e) FQT Test Plan and Test Procedures.

f) Any support documents required during FQT.

A flow chart showing a representative sequence of activities for internal configuration control is shown in Figure 5-8.

### 5.3.5 Maintenance of Internally Controlled Documentation

After CDR, copies of the CPCI Product Specifications should be kept current for the benefit of programmers and qualification test procedure writers. In addition, a completely updated Product Specification should be available at the start of FQT. Test plans and procedures also should be kept current during this period of internal control.

Contractor forms such as the Document Update Transmittals (DUTs) described in Table 5-1 usually are used for updates of all documents subject to internal control.

### 5.3.6 Maintenance of Internally Controlled Software

Contractor standards and methods for maintaining software configuration integrity during FQT should be essentially the same as those used for baseline configuration control. Anything less would compromise the validity of the qualification test process. Software configurations should be carefully controlled during PQT tests also.

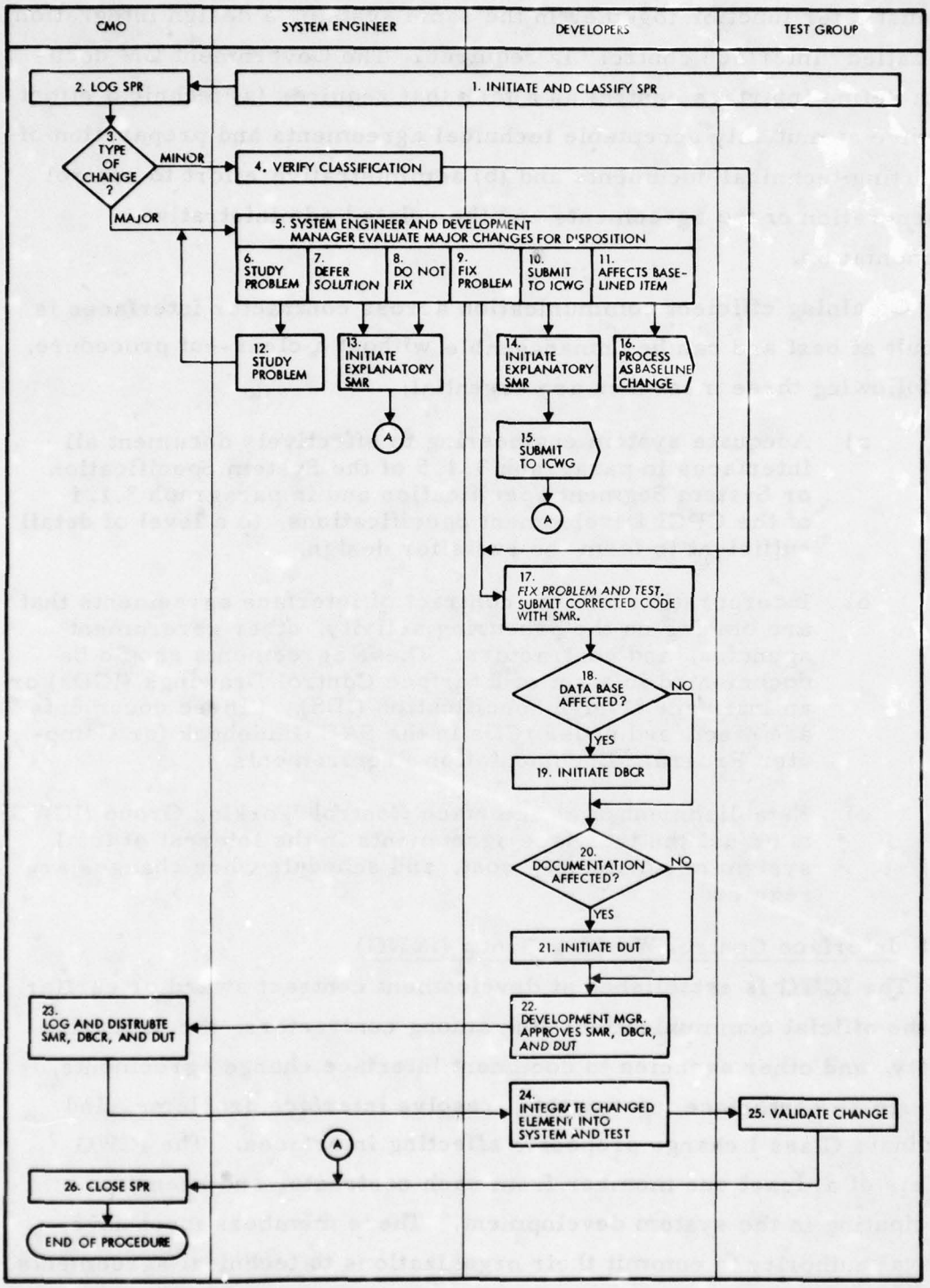



Figure 5-8. Contractor Internal Configuration Control Flow Chart

## 5.4 INTERFACE CONTROL

When two or more CPCIs not being developed by the same contractor must later function together in the same system, a design integration task called "interface control" is required. The Government CM documents define interface control as a task that requires (a) technical effort to arrive at mutually acceptable technical agreements and preparation of supporting technical documents and (b) administrative effort to control the generation of the agreements and the related administrative documentation.

Obtaining efficient communication across contractor interfaces is difficult at best and can be unmanageable without a clear-cut procedure. The following three measures are essential:

a) Adequate system engineering to effectively document all interfaces in paragraph 3.1.5 of the System Specification or System Segment Specification and in paragraph 3.1.1 of the CPCI Development Specifications, to a level of detail sufficient to form the basis for design.

b) Incorporation into the contract of interface agreements that are binding on the procuring activity, other government agencies, and contractors. These agreements should be documented in a set of Interface Control Drawings (ICDs) or an Interface Design Specification (IDS). (These documents are discussed under ICDs in the SAE Guidebook for Computer Program Documentation Requirements.)

c) Establishment of an Interface Control Working Group (ICWG) to adjust the interface agreements in the interest of total system performance, cost, and schedule when changes are required.

### 5.4.1 Interface Control Working Group (ICWG)

The ICWG is established at development contract award or earlier. It is the official communications link among contractors, the procuring activity, and other agencies to document interface change agreements, exchange new interface information, resolve interface problems, and coordinate Class I change proposals affecting interfaces. The ICWG consists of at least one member from each contractor and agency participating in the system development. These members must have approval authority to commit their organizations to technical agreements. Contractor representatives usually are system engineers. The procuring

activity, prime contractor, or integrating contractor provides the ICWG chairman. The chairman or his designee prepares the agenda and records the minutes and action items.

5.4.2 Interface Change Processing

A combined Interface Change Request/Notice (ICR/ICN) is shown in Figure 5-9. This form is used as either an ICR or an ICN, with the usage indicated by a check in the appropriate box at the top. A procedure for interface control using this form is shown in Figure 5-10 and is described in the following paragraphs.

An interface change is initiated by a contractor (step 1) via an Interface Change Request (ICR) and is classified by the originator as to priority:

a) Emergency. ICWG meeting to be held not later than 48 hours after chairman receives ICR.

b) Urgent. ICWG meeting to be held within two weeks.

c) Routine. ICWG meeting to be held within 30 days.

The originator sends the ICR to his project CCB for approval. After CCB approval, the ICR is forwarded to the ICWG.

The ICWG chairman places new ICRs on the agenda for the next scheduled meeting (step 2). If the ICR has an emergency or urgent priority, a special meeting is called. For routine ICRs, a preliminary agenda is sent to members 10 days before the meeting.

When the ICWG meets, it first reviews all outstanding ICRs (step 3) to update their status in regard to previously assigned actions, to decide the appropriate dispositions (steps 4, 6, 8, 11, or 13), and to record the dispositions on Interface Change Notices (ICNs). New ICRs are then reviewed and dispositions are made.

| INTERFACE CHANGE | | ☐ REQUEST<br>☐ NOTICE |
|---|---|---|
| TO: | | CPIC CONTROL NUMBER |
| FROM:<br>NAME _____ PHONE _____<br>FIRM _____ DATE _____ | | DATE LOGGED<br>ORIG. CONTROL NUMBER |
| DOCUMENT AFFECTED | CONFIGURATION/MODEL | EFFECTIVE DATE |
| REASON FOR CHANGE | | |
| DESCRIPTION OF CHANGE | | |

CONCURRENCE:

| | ORGANIZATION | SIGNATURE | DATE |
|---|---|---|---|
| CPIC | | | |
| REVIEWER 1 | | | |
| REVIEWER 2 | | | |
| GSE/TD | | | |
| APPROVAL AGENCY | | | |

Figure 5-9. Interface Change Request/Notice (ICR/ICN)

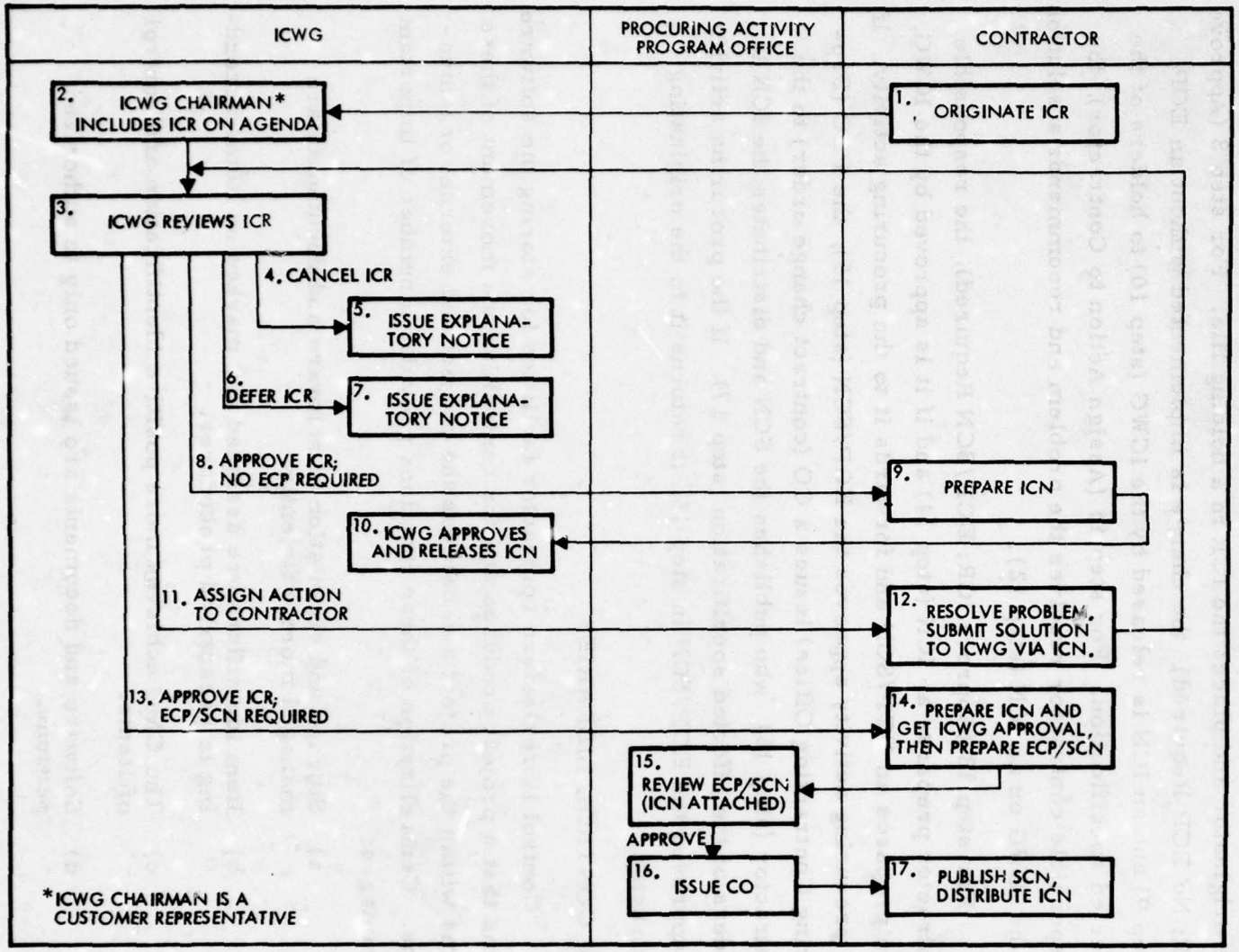



Figure 5-10. ICWG Interface Control Flow Chart

In the case of step 4 (Cancel ICR), the ICWG initiates an explanatory ICN to notify the originator about the cancellation. For step 6 (Defer ICR Until Later Date), the ICWG issues an explanatory ICN to the originator and places the ICR in a holding file. For step 8 (Approve ICR; No ECP Required), the change is implemented without an ECP (step 9) and an ICN is released by the ICWG (step 10) to holders of the affected specification. For step 11 (Assign Action to Contractor), the responsible contractor resolves the problem and recommends a solution to the ICWG on an ICN (step 12).

For step 13 (Approve ICR; ECP/SCN Required), the responsible contractor prepares an ICN (step 14) and if it is approved by the ICWG, also prepares an ECP/SCN and forwards it to the procuring activity. If the procuring activity approves the ECP/SCN (step 15), the PCO (Procuring Contracting Office) issues a CO (contract change order) to the contractor (step 16), who publishes the SCN and distributes the ICN to holders of the affected specification (step 17). If the procuring activity disapproves the ECP/SCN in step 15, it returns it to the originating contractor.

## 5.5 CONTROL LIBRARIES

Control libraries are contractor facilities for storing the software items that a project produces and for controlling the movement of these items within the project and between the project and external organizations. Centralization of these functions provides a number of important advantages:

a) Storage and circulation of software and documents are managed more efficiently.

b) Item identifiers are assigned and marked on items according to prescribed procedures.

c) The CMO achieves more positive identification and control of items.

d) Software and documents are issued only to authorized persons.

Software assembly and maintenance personnel responsible for assembling and updating master libraries of source and object code and related tasks generally use these control libraries for storage of their tapes, decks, discs, and listings. Sometimes the software assembly and maintenance function and the control library function will be performed by the same contractor personnel.

# 6. SPECIFIC GUIDANCE FOR CONFIGURATION STATUS ACCOUNTING

Configuration status accounting (CSA) methods for baseline configuration control and contractor internal configuration control are described in this section. CSA responsibilities of program participants and use of automated CSA techniques also are discussed.

## 6.1 RESPONSIBILITIES FOR CONFIGURATION STATUS ACCOUNTING

All commands and all contractors participating in system acquisition or deployment have configuration status accounting responsibilities consistent with their functional roles. These responsibilities generally take the following form:

a) AFSC Procuring Activity. As the implementing agency, the AFSC procuring activity must itself assume primary responsibility for all pre-PMRT CSA requirements or assign those responsibilities to a contractor. When a number of Government agencies and contractors are involved in an acquisition program, one of the contractors usually is chosen for this task, which some Government CM documents refer to as CSA integration.

b) CSA Integrator. The CSA integrator (either a contractor or the procuring activity) is responsible for selecting CSA data elements, tailoring record and report formats, establishing the frequency of reports, maintaining records, and issuing a Configuration Item Index (CII) and Configuration Status Accounting Reports (CSARs) for baselined CPCIs. The integrator should design the index and report formats to assist the contractual and engineering tasks of system acquisition and should coordinate the formats with all other program participants who will be using them. This coordination is especially important for reports that are to be automated. The CSA integration task is transferred to AFLC at PMRT.

c) AFLC Air Logistics Center (ALC). As the supporting agency for a system, the ALC is responsible for all post-PMRT CSA tasks. AFLC CSA indexes and reports are designed to assist in the maintenance and logistics support of the system, emphasizing TCTO actions against systems and items in the Government inventory.

d) Other Contractors and Agencies. Other contractors and agencies participating in pre-PMRT activities are required to record their own CSA data and submit inputs to the CSA integrator at required intervals. Data concerning accomplishment of approved changes to CPCIs and hardware CIs in

their custody should be included. Data may be in the form of the contractor's or agency's own CSA system reports if they meet program requirements. A representative set of contractor CSA logs and reports and their relationship to the configuration control forms is shown in Figure 6-1.

## 6.2 BASELINE CONFIGURATION STATUS ACCOUNTING

### 6.2.1 Baseline CSA Reporting Documents

CSA documents are major factors in the effectiveness of a program's CM system. Initially they record the establishment of configuration identification baselines, and as the program progresses, record and report the status of proposed changes to each CPCI and hardware CI and the implementation progress of approved changes.

Two basic kinds of CSA reporting documents are used. One, a configuration index, defines the current approved configuration of a configuration item or system. The other, a change status report, gives the implementation status of approved changes to the configuration item or system.

Prior to the product baseline, the two CSA documents address the configuration status of CPCIs or hardware CIs and have the following characteristics:

a) Configuration Index (DID DI-E-3122). This document, produced by the development contractor, reports the current status of configuration item development in terms of specifications and other documents that depend on the configuration, such as qualification Test Plans and Procedures, User Manuals, and the Version Description Document. It lists all ECPs and SCNs incorporated, approved ECPs not yet incorporated, and other data.

b) Change Status Report (DID DI-E-3123). This report, also prepared by the development contractor, provides the detailed status of all proposed and approved ECPs to the documents listed in the Configuration Index. This report always is issued with the Configuration Index.

After the product baseline is established for a configuration item, these documents no longer are issued. Instead, a new pair of CSA documents address the configuration status of the entire system:

a) Configuration Identification Index (CII) (DID DI-E-3133). This index lists the CPCIs and hardware CIs making up the

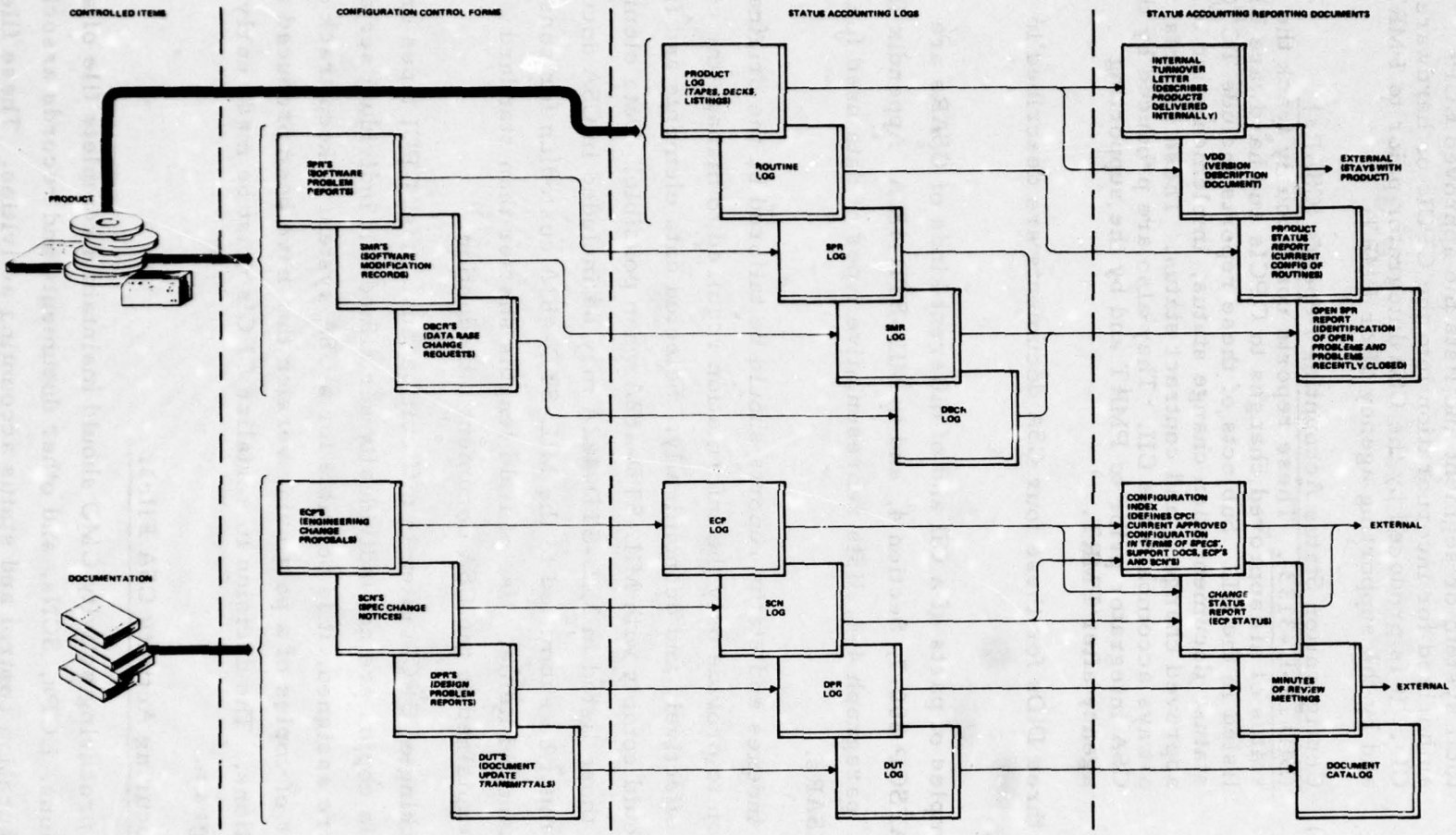



Figure 6-1. Relationship of Contractor Status Accounting Documents to Configuration Control Forms

total system or segment and lists the approved ECPs authorized for incorporation into the CPCIs or hardware CIs. It is produced by the CSA integrator prior to PMRT and by the supporting agency after PMRT.

b) Configuration Status Accounting Reports (CSARs) (DID DI-E-3133). These reports continuously track the status of all approved changes to CPCIs and hardware CIs listed in the CII. Subjects of these reports include TCTO status, documentation change status, implementation of approved changes, and contract status. These reports always accompany the CII. They also are produced by the CSA integrator prior to PMRT and by the supporting agency after PMRT.

The three DIDs for these four CSA documents are described in Table 3-1.

Examples of parts of a CII and of different kinds of CSARs are shown in AFSCP 800-7, Section 4, and in MIL-STD-482A, Appendix III. AFR 65-3, paragraph 4-4, lists representative types of data used for CIIs and CSARs.

CSA indexes and status reports should be tailored by the originating organization to provide only the information required to manage the configuration effectively and economically. Selected data elements and field lengths should comply with MIL-STD-482A when possible. Data elements other than those listed in MIL-STD-482A may be included in CSA documents but must be submitted to the MIL-STD-482A custodian for consideration as standard items. Use of field lengths shorter than standard should be explained in the CSA document introduction.

Tracking of CPCIs often is difficult because most CPCI tapes or other media copies are not individually serialized. If individual serial numbers are assigned, it is possible for a CSA system to keep track of the number of copies of a particular version that have been produced and their locations. The decision to serialize CPCIs must be made early in the program.

### 6.2.2 Procuring Activity CSA Files.

The procuring activity CMO should maintain a complete file of all specifications, ECPs, SCNs, and other documents and records associated with configuration control and status accounting activities. These files

provide the traceability for a CPCI and its changes that is required for program management. They also provide part of the PMRT package to be transferred to the supporting agency.

## 6.3 CONTRACTOR INTERNAL CONFIGURATION STATUS ACCOUNTING

### 6.3.1 Contractor Internal CSA Documents

A representative set of internal CSA reporting documents is shown in the right-hand column of Figure 6-1, together with the baseline CSA reporting documents. The internal CSA documents are as follows:

a) Product Status Report. Lists current information for software products. Typical information includes (1) computer program/component identification, (2) revision/version/modification status, (3) baseline identifiers, (4) location, if software item is used at different locations, (5) reference to outstanding/unincorporated problem reports, and (6) reference to changes incorporated in latest version of the software item (by user location if applicable).

b) Open SPR Report. Identifies all open problem reports and those closed since the preceding report. Typical information includes (1) problem report number, (2) date initiated, (3) originator, (4) associated change report number, if any, (5) associated ECP, if any, (6) identification and modification level, (7) computer program/component identification, and (8) status (accept, reject, open, closed, etc.).

c) Document Catalog. Defines the current status of all deliverable technical documents. Usually includes (1) document number and date, (2) document title, (3) reference to change requests, and (4) revision status.

d) Internal Turnover Letter. This document is used within a contractor's organization to identify a software item that is formally transferred between development areas, such as from the development group to the integration test group. It can be considered a status accounting document because its contents are based on information in one or more status accounting logs. Its contents typically include (1) a list of the routines being turned over, together with their version identification, (2) a list of all changes and corrections incorporated in the routines, and (3) a list of all known problems remaining in the routines and the status of probem correction.

Software change status information can either be included in the Open SPR Report or placed in a separate report. A separate Software Change Status Report might include the following items: (1) change report number, (2) date initiated, (3) originator, (4) associated problem report number, if any, (5) associated ECP, if any, (6) identification and modification level, (7) computer program/component identification, and (8) status (accept, reject, open, closed, etc.).

### 6.3.2 Contractor CSA Logs and Files

Contractor CMOs should record problem and change transactions on a series of logs that contain sufficient cross-reference data to permit convenient tracking of problems and changes. A set of contractor logs for both baseline and internal CSA purposes is shown in Figure 6-1. The CMOs also should maintain a complete file of specifications, ECPs, SCNs, and internal control forms.

## 6.4 AUTOMATED CONFIGURATION STATUS ACCOUNTING

DODD 5010.19 stipulates that "automation of status accounting shall be employed only when the volume of data or rapid response time makes it necessary, and is economically feasible." AFSCP 800-7, Chapter 4, has considerable material on this subject.

# APPENDIX A

## GLOSSARY

Allocated Configuration Identification (ACI). Current, approved performance oriented specifications governing the development of configuration items that are part of a higher level CI, in which each specification (1) defines the functional characteristics that are allocated from those of the higher level CI, (2) establishes the tests required to demonstrate achievement of its allocated functional characteristics, (3) delineates necessary interface requirements with other associated configuration items, and (4) establishes design constraints, if any, such as component standardization, use of inventory items, and integrated logistic support requirements. (DODI 5010.21)

Baseline. A configuration identification document or a set of such documents formally designated and fixed at a specific time during a CI's life cycle. Baselines, plus approved changes from those baselines, constitute the current configuration identification. For configuration management there are three baselines, as follows:

a) Functional Baseline. The initial approved functional configuration identification.

b) Allocated Baseline. The initial approved allocated configuration identification.

c) Product Baseline. The initial approved or conditionally approved product configuration identification. (DODI 5010.21)

Computer Data. Basic elements of information used by computer equipment in responding to a computer program. Data operated on, produced by, or otherwise used by a computer program. (AFSC Supplement 1 to AFR 800-14, Volume I)

Computer Program. A series of instructions or statements in a form acceptable to an electronic computer, designed to cause the computer to execute an operation or operations. (AFR 800-14, Volume I)

Computer Program Component (CPC). A functionally or logically distinct part of a computer program configuration item (CPCI)

distinguished for purposes of convenience in designing and specifying a complex CPCI as an assembly of subordinate elements. (MIL-STD-483)

Computer Program Configuration Item (CPCI). See "Configuration Item."

Computer Software. Computer programs and/or computer data.

Configuration. The functional and/or physical characteristics of hardware/software as set forth in technical documentation and achieved in a product. (DODI 5010.21)

Configuration Control. The systematic evaluation, coordination, approval or disapproval, and implementation of all approved changes in the configuration of a CI after formal establishment of its configuration identification. (DODI 5010.21)

Configuration Identification. The current approved or conditionally approved technical documentation for a configuration item as set forth in specifications, drawings and associated lists, and documents referenced therein. (DODI 5010.21)

Configuration Item (CI). An aggregation of hardware/software, or any of its discrete portions, which satisfies an end use function and is designated by the Government for configuration management. CI's may vary widely in complexity, size and type, from an aircraft, electronic or ship system to a test meter or round of ammunition. During development and initial production, CI's are only those specification items that are referenced directly in a contract (or an equivalent in-house agreement). During the operation and maintenance period, any reparable item designated for separate procurement is a configuration item. (DODI 5010.21).

Configuration Management (CM). A discipline applying technical and administrative direction and surveillance to (1) identify and document the functional and physical characteristics of a configuration item, (2) control changes to those characteristics, and (3) record and report change processing and implementation status. (DODI 5010.21) CM also verifies that a completed configuration item and its documentation meet contractual requirements.

Configuration Status Accounting (CSA). The recording and reporting of the information that is needed to manage configuration effectively, including a listing of the approved configuration identification, the status of proposed changes to configuration, and the implementation status of approved changes. (DODI 5010.21)

Deficiencies. Deficiencies consist of two types: (1) conditions or characteristics in any hardware/software which are not in compliance with specified configuration, or (2) inadequate (or erroneous) configuration identification which has resulted, or may result, in configuration items that do not fulfill approved operational requirements. (DODI 5010.21)

Deviation. A specific written authorization, granted prior to the manufacture of any item, to depart from a particular performance or design requirement of a contract, specification, or referenced document, for a specific number of units or specified period of time. (DODI 5010.21)

Form, Fit and Function. That configuration comprising the physical and functional characteristics of the item as an entity but not including any characteristics of the elements making up the item. (DODI 5010.21)

Functional Characteristics. Quantitative performance, operating and logistic parameters and their respective tolerances. Functional characteristics include all performance parameters, such as range, speed, lethality, reliability, maintainability, safety. (DODI 5010.21)

Functional Configuration Audit (FCA). The formal examination of functional characteristics test data for a configuration item, prior to acceptance, to verify that the item has achieved the performance specified in its functional or allocated configuration identification. (DODI 5010.21)

Functional Configuration Identification (FCI). The current approved technical documentation for a configuration item which prescribes (1) all necessary functional characteristics, (2) the tests required to demonstrate achievement of specified functional characteristics, (3) the necessary interface characteristics with associated CI's, (4) the CI's key functional characteristics and its key lower level CI's, if any, and

(5) design constraints, such as envelope dimensions, component standardization, use of inventory items, integrated logistics support policies. (DODI 5010.21)

Key Functional Characteristics. Those functional characteristics that critically affect the configuration item's satisfactory fulfillment of the operational requirements; for example, a transport aircraft's payload/range characteristics. (DODI 5010.21)

Physical Characteristics. Quantitative and qualitative expressions of material features, such as composition, dimensions, finishes, form, fit, and their respective tolerances. (DODI 5010.21)

Physical Configuration Audit (PCA). The formal examination of the "as-built" configuration of a unit of a CI against its technical documentation in order to establish the CI's initial product configuration identification. (DODI 5010.21)

Product Configuration Identification (PCI). The current approved or conditionally approved technical documentation which defines the configuration of a CI during the production, operation, maintenance, and logistics support phases of its life cycle, and which prescribes (1) all necessary physical or form, fit and function characteristics of a CI, (2) the selected functional characteristics designated for production acceptance testing, and (3) the production acceptance tests. (DODI 5010.21)

Specification. A document that clearly and accurately describes the essential technical requirements for an item, material, or service and the procedures for determining that the requirements have been met.

Waiver. A written authorization to accept a configuration item or other designated items, which during production or after having been submitted for inspection, are found to depart from specified requirements, but nevertheless are considered suitable for use "as is" or after rework by an approved method. (DODI 5010.21)

## APPENDIX B

## OUTLINE FOR CONTRACTOR SOFTWARE CONFIGURATION MANAGEMENT PLAN (COMPLIES WITH MIL-STD-483, APPENDIX I)

### 1.0 INTRODUCTION

This section shall define the purpose and scope of the Configuration Management Plan.

### 2.0 ORGANIZATION AND RESPONSIBILITIES

This section shall identify the project organizational unit that will perform configuration management, describe the authority and responsibilities of this unit, and describe the interfaces between this unit and other organizations within and external to the project. It also shall describe the contractor's configuration management policies and practices in sufficient detail to establish their effectiveness.

### 3.0 CONFIGURATION IDENTIFICATION

#### 3.1 Configuration Identification Documents

This subsection shall identify the configuration identification documents that will be prepared and delivered and state the applicability of specific portions of MIL-STD-490, MIL-STD-483, or other contractual compliance documents. It also shall discuss the authority and responsibilities of the contractor and procuring activity in establishing CPCI configuration identifications and changes to those identifications and the responsibility for cost and schedule impacts resulting from changes. It also shall state any limitations on delivery to, or usage by, the procuring activity.

#### 3.2 Configuration Identification Baselines

This subsection shall identify the configuration identification baselines to be employed by the project, as defined in the contract SOW. The following shall be defined for each baseline:

a) Products included in the baseline (Development Specifications, Product Specifications, CPCIs, etc.)

b) Review and approval events associated with the baseline.

c) Method of establishing the baseline.

3.3 Item Identifiers

This subsection shall specify the types of products to be identified and the rules for assigning identifiers and marking the products.

4.0 CONFIGURATION CONTROL

4.1 Control Procedures

This subsection shall specify the procedures for both baseline configuration control and contractor internal configuration control. For each control procedure, describe:

a) Products subject to the procedure.

b) Review, approval, and implementation sequence for problem reports/change proposals.

c) Review and approval authorities (e.g., CCB, ICWG, Project Manager).

Include a diagram relating project activities and events in the development cycle to the evolving software products and showing for each controlled product:

a) Period of control.

b) Degree of control applied during each period (e.g., control internal to the project, control by procuring activity concurrence, control by procuring activity approval).

c) Delivery events (internally to the internal configuration control environment and externally to the procuring activity).

Include a discussion of the method to be used for communicating control matters between the contractor and procuring activity. Also discuss technical interface control, both between the contractor and procuring activity and, when appropriate, between the contractor and other participating contractors.

State the applicability of DOD-STD-480A, MIL-STD-481, MIL-STD-483, and any other compliance documents.

4.2 Storage and Release

Describe the methods for the formal controlled storage and release of software master tapes and document master copies. Describe the operations of the project's product control library, the provisions for storage and release, and the procedure for distribution.

5.0 CONFIGURATION STATUS ACCOUNTING

This section shall state the contractor's understanding of his specific contractual role and responsibility for configuration status accounting. This includes whether he is to (a) submit data to an integrating agency or contractor who will prepare and distribute reports, (b) prepare and distribute the reports himself, or (c) accept data from other contractors and participating agencies, collate such data with his own, and prepare and distribute the reports.

This section also shall describe an appropriate configuration status accounting system for meeting these responsibilities, including the records and reports required to provide traceability of change proposals, approved changes, and implemented changes to controlled items.

This section also shall describe the format, content, intended use, distribution, processing, and retention for each record and report to be prepared, including a configuration index and a change status report. Any automated status accounting techniques intended shall be described, and any limitations on procuring activity requests for changing initial formats shall be stated.

Applicability of MIL-STD-482A also shall be stated.

6.0 SUBCONTRACTOR/VENDOR CONTROL

If applicable, this section shall describe the controls that will be employed to enforce subcontractor and/or vendor adherence to project configuration management standards and procedures. This shall include explanation of the methods employed to determine subcontractor or vendor CM capabilities.

## 7.0 PROGRAM PHASING

This section shall define the major CM milestones, including:

a) Establishment of the project Configuration Control Board (CCB).

b) Milestones related to the preparation and maintenance of specifications.

c) Establishment of configuration identifications.

d) Establishment of control agreements with associate contractors.

e) Establishment of configuration status accounting procedures.

## 8.0 MANAGEMENT INTEGRATION OF CM

This section shall describe the relationship of CM with other project management activities. This includes the relationship of CPCI-level CM to the project work breakdown structure and the relationships between major CM events and other critical project events.

## 9.0 CONFIGURATION AUDITS

This section shall describe plans for conducting or supporting FCAs, PCAs, and FQRs. It shall define the software items and documents to be audited, the auditing authority, the method for handling deviations and waivers, the change forms and procedures, and the method for numbering changes.

## APPENDIX C

## BIBLIOGRAPHY OF GOVERNMENT DOCUMENTS

### A. DOD DOCUMENTS

| | | |
|---|---|---|
| DOD Directive 5000.1 | 18 January 1977 | Major System Acquisitions |
| DOD Directive 5000.2 | 18 January 1977 | Major System Acquisition Process |
| DOD Directive 5000.29 | 26 April 1976 | Management of Computer Resources in Major Defense Systems |
| DOD Directive 5010.19 | 17 July 1968 | Configuration Management |
| • Change 1 | 6 August 1968 | |
| • Change 2 | 7 April 1970 | |
| DOD Instruction 5010.21 | 6 August 1968 | Configuration Management Implementation Guidance |

### B. USAF DOCUMENTS

| | | |
|---|---|---|
| AF Regulation 57-4 | 15 December 1977 | Modification Program Approval |
| • Interim Message Change No. 78-1 | 9 February 1978 | |
| AF Regulation 65-3 | 1 July 1974 | Configuration Management |
| • Change 1 | 1 September 1974 | |
| • AFSC Supplement 1 | 25 July 1975 | |
| AF Regulation 80-14 | 10 February 1975 | Test and Evaluation |
| • AFSC Supplement 1 | 16 June 1975 | |
| AF Regulation 800-2 | 16 March 1972 | Program Management |
| • AFSC Supplement 1 | 18 October 1974 | |
| • SAMSO Supplement 1 | 11 February 1977 | |
| AF Regulation 800-4 | 10 March 1975 | Transfer of Program Management Responsibility |
| • AFSC/AFLC Supplement 1 | 14 August 1975 | |
| • SAMSO Supplement 1 | 6 April 1976 | |
| AF Regulation 800-14, Vol. I | 12 September 1975 | Management of Computer Resources in Systems |
| • AFSC Supplement 1 | 8 August 1977 | |
| AF Regulation 800-14, Vol. II | 26 September 1975 | Acquisition and Support Procedures for Computer Resources in Systems |
| • AFLC Supplement 1 | 18 October 1976 | |

| | | |
|---|---|---|
| AF Regulation 800-19 | 27 May 1975 | System or Equipment Turnover |
| **C. AFSC DOCUMENTS** | | |
| AFSC Pamphlet 800-3 | 9 April 1976 | A Guide for Program Management |
| AFSC Pamphlet 800-7 | 1 December 1977 | Configuration Management |
| **D. SAMSO DOCUMENT** | | |
| SAMSO Phamphlet 74-2 | 1 September 1976 | Contractor Quality Assurance Evaluation |
| **E. TECHNICAL ORDERS** | | |
| T.O. 00-5-15<br>• Change 1 | 15 January 1977<br>15 December 1977 | Air Force Time Compliance Technical Order System |
| **F. SPECIFICATIONS** | | |
| DOD-D-1000B | 28 October 1977 | Drawings, Engineering and Associated Lists |
| MIL-P-83497 (USAF) | 16 August 1976 | Procedures for Preparation of Programmed Tapes and Cards |
| MIL-S-52779 (AD) | 5 April 1974 | Software Quality Assurance Program Requirements |
| MIL-S-83490 | 30 October 1968 | Specifications, Types and Forms |
| MIL-T-38804 (USAF)<br>• Amendment 2 | 31 July 1972<br>1 November 1977 | Time Compliance Technical Orders (TCTOs), Preparation of |
| **G. STANDARDS** | | |
| MIL-STD-100B<br>• Notice 1<br>• Notice 2 | 15 October 1975<br>20 February 1976<br>15 April 1976 | Engineering Drawing Practices |
| MIL-STD-130E | 5 August 1977 | Identification Marking of U.S. Military Property |
| DOD-STD-480A | 12 April 1978 | Configuration Control - Engineering Changes, Deviations, and Waivers |
| MIL-STD-481A | 18 October 1972 | Configuration Control - Engineering Changes, Deviations, and Waivers (Short Form) |

| | | |
|---|---|---|
| MIL-STD-482A | 1 April 1974 | Configuration Status Accounting Data Elements and Related Features |
| MIL-STD-483 (USAF)<br>• Notice 1 | 31 December 1970<br>1 June 1971 | Configuration Management Practices for Systems, Equipment, Munitions, and Computer Programs |
| MIL-STD-490<br>• Notice 1 | 30 October 1968<br>1 February 1969 | Specification Practices |
| MIL-STD-1521A (USAF) | 1 June 1976 | Technical Reviews and Audits for Systems, Equipments, and Computer Programs |

# APPENDIX D

## EXAMPLES OF CONTRACTOR CONFIGURATION CONTROL FORMS

PROJECT TITLE
ACRONYM

# DESIGN PROBLEM REPORT

**DPR** TO: CONFIGURATION MANAGEMENT PAGE 1 OF ____

② DPR NUMBER ____
DATE LOGGED ____
OTHER CHG. CONTROL NOS. ____

① FROM:
NAME ____ DATE ____

③ DOCUMENT/ROUTINE: ____ ID ____ VERSION/REVISION ____
REVIEWER/AGENCY: ____
PAGE/SECTION/FIGURE: ____

④ PROBLEM: ____

⑤ RECOMMENDATION: ____

⑥ INCORPORATION APPROVAL:
PRODUCT ASSURANCE ____ DATE ____
PROJECT MANAGER ____ DATE ____
TEST MANAGER ____ DATE ____
DEVELOPMENT MANAGER ____ DATE ____

RECOMMENDED PRIORITY ☐ LOW ☐ MED ☐ HIGH ☐ PRODUCT IMPROVEMENT ☐ REJECT

⑦ CHANGE INCORPORATED BY: SMR NO. ____ SPMR NO. ____ DUT NO. ____ ECP NO. ____
DBCR NO. ____ CCR NO. ____ SCN NO. ____

⑧ NEW ID(S) OF CHANGED ITEM(S) NAME ____ VERSION/REVISION ____
____ ____
____ ____

⑨ DISPOSITION: DOC NO. ____
INCORP. IN REEL/DISK NO. ____ BY ____ DATE ____
BLOCK/BUILD/LOOP ____ VERIF BY CMO ____

PROJECT TITLE
ACRONYM

# DOCUMENTATION UPDATE TRANSMITTAL

**DUT** TO: CONFIGURATION MANAGEMENT PAGE 1 OF ____

② DUT NUMBER ____
DATE LOGGED ____
ORIG SPR/DPR NUMBER ____

① FROM:
NAME ____ DATE ____

③ UPDATES TO: BLOCK ____ BUILD ____ LOOP ____
☐ DESIGN SPEC. (B5) ☐ PRODUCT SPEC. (C5) ☐ TEST SPEC ☐ TEST PLAN/ PROCEDURE ☐ OTHER

④ DOCUMENT TITLE ____ DOCUMENT IDENT. NO. ____
LAST DUT REFLECTED AGAINST THIS DOCUMENT WAS ____
IMPACTS ROUTINE(S) ____ SUBSYSTEM ____

⑤ JUSTIFICATION FOR UPDATE:
DPR NO. ____ SPR/SMR NO. ____ SCN NO. ____
OTHER (EXPLAIN) ____

⑥ UPDATED PAGES (ATTACHED): ____

⑦ REVIEW AND APPROVAL:
APPROVAL (DEVELOPMENT MANAGER) ____ DATE ____
CONCURRENCE (CONFIGURATION MANAGEMENT) ____ DATE ____
COMMENTS/STATUS: ____

⑧ ERROR CATEGORY OF PROBLEM CAUSE (SEE BACK) ____, ____, ____, ____, ____, ____

⑨ DATE INCORPORATED ____ VERIFIED BY CMO ____
DOC. NO. ____ REV ____ CHG ____

PROJECT TITLE
ACRONYM

# SOFTWARE PROBLEM REPORT

**SPR** TO: CONFIGURATION MANAGEMENT

② SCMO SPR NUMBER _____

① FROM:
NAME _____ DATE _____

DATE LOGGED _____

③ PROBLEM DISCOVERED DURING:
☐ DESIGN ☐ DEV ☐ UNIT TEST ☐ SYSTEM INTEG ☐ ACCEPT TEST ☐ FCA/PCA ☐ OPER/MAINT

④ PRIORITY: ☐ HIGH ☐ MED ☐ LOW

⑤ CONFIGURATION: OPER SYS _____ DEV MASTER ID _____ TEST CASE ID _____
DATA BASE ID _____ DATA BASE REEL NO. _____
HARDWARE CONFIG _____ CI/CPCI NO. _____

⑥ PROBLEM WITH: ☐ ROUTINE ☐ DOCUMENT ☐ DB ☐ COMPOOL ☐ DESIGN ☐ TEST
ROUTINE NAME _____ ID _____ VERSION/REVISION _____
DOCUMENT TITLE _____ DOC NUMBER/REV _____

⑦ PROBLEM DESCRIPTION: _____

⑧ EFFECTS/IMPACT/SPECIAL NOTES: _____

⑨ APPROVED FOR REVIEW ☐ REJECTED ☐ ASSIGNED TO _____
REVIEWER/COMMENTS _____

NAME _____ ORGANIZATION _____ DATE _____

⑩ ACTION REQUIRED: ASSIGNED TO: _____

| SMR NO.(S) | DPR NO.(S) | CCR NO.(S) | DBCR NO.(S) | DUT NO.(S) | ECP/SCN NO.(S) |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

1

PROJECT TITLE
ACRONYM

# SOFTWARE MODIFICATION RECORD

**SMR** TO: CONFIGURATION MANAGEMENT PAGE 1 OF ____

② SMR NUMBER ____
DATE LOGGED ____
ORIG SPR. NUMBER ____

① FROM:
NAME ____ DATE ____

③ MODIFICATION SUBMITTED FOR: SUBSYSTEM ____ CI/CPCI NO. ____
PROGRAM NAME ____ ID ____ VERSION/REVISION ____
DOCUMENT TITLE ____ DOC/NO/REV ____
RESPONSE TO OTHER SPR(S) ____ SUPPLEMENTS SMR(S) ____

④ REASON FOR SUBMITTAL: ☐ DELIVERY/TURNOVER ☐ SPR EXPLANATION ☐ CODE MODIFICATION
PROBLEM ANALYSIS: ____

⑤ CHANGE DESCIRPTION: ____

ANALYST ____ DATE ____

⑥ CHANGE IMPACT: ERROR CATEGORY(IES)(SEE BACK) ____
CCR NO. ____ DBCR NO. ____ DUT NO. ____ ECP/SCN NO. ____ DPR NO. ____

⑦ SUBMITTAL INSTRUCTIONS FOR PROGRAM DELIVERIES/MODIFICATIONS:
DISK NO. ____ REEL NO. ____ INPUT ID ____ VERSION/REVISION ____ SOURCE TYPE ____

⑧ VERIFICATION/AUTHORIZATION:
MODIFICATION(S) ☐ HAVE ☐ HAVE NOT BEEN TESTED AT ____ BY ____ DATE ____
SMR APPROVED BY ____ DATE ____ QA OR EQUIVALENT ____ DATE ____

⑨ DISPOSITION:
INCORP REEL/DISK NO. ____ BY ____ DATE ____ VERIF. BY CMO ____

PROJECT TITLE
ACRONYM

# DATA BASE CHANGE REQUEST

**DBCR** TO: CONFIGURATION MANAGEMENT

(1) FROM: _____ DATE: _____

FOR DB: ☐ DEV ☐ TEST ☐ OPR

CATEGORY: ☐ DATA ☐ OVERRIDES ☐ ERROR MESSAGE

(5) CHANGE ACTION: ☐ ADD ☐ CHANGE ☐ DELETE

BLOCK NAME _____ BASE DBID _____

OTHER _____ S/W BLK _____

(2) MESSAGE NUMBER

(3) CHANGE REVIEWS/APPROVALS

PROJECT MANAGER _____ DATE _____

DEVELOPMENT MGR _____ DATE _____

TEST MANAGER _____ DATE _____

(WHEN APPLICABLE)

INTEGRATION MGR _____ DATE _____

CUSTOMER _____ DATE _____

(4) DBCR NO. _____

DATE LOGGED _____

ORIG. SPR/DPR NO. _____

(6) FINAL ACTION

NEW DBID: _____

DATE VERIF: _____

VERIF BY: _____

(7) REMARKS:

(8)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80

(9) ERROR CATEGORY OF PROBLEM CAUSE (SEE BACK) _____, _____, _____, _____, _____, _____, _____,

☆U.S. Government Printing Office: 1979 — 657-002/146